QUERY 3; A DATA BASE INQUIRY SYSTEM DESCRIPTION AND USER'S TUTORIALS

Leo Joseph Faneuf

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

QUERY 3; A DATA BASE INQUIRY SYSTEM

DESCRIPTION AND USER'S TUTORIALS

by

Leo Joseph Faneuf III

March 1980

Thesis Advisor: G.K. Poock

Approved For Public Release; Distribution Unlimited

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) QUERY 3; A DATA BASE INQUIRY SYSTEM DESCRIPTION AND USER'S TUTORIALS		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis March 1980 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Leo Joseph Faneuf		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE March 1980 13. NUMBER OF PAGES 99
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Query 3
Data Base Inquiry
Data Base Management System

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This work examines a data base inquiry system called Query 3. A demonstration version of this experimental system is available on the Advanced Research Projects Agency Network (ARPANET) at Computer Host 116.
This research contains a description of the system, and basic and advanced tutorial user's guides. System description covers the various aspects of Query 3 including it's online system (NLS)

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Block 20 continued.

base, ARPANET operations, human interface capabilities, system operation, and data structure. The basic tutorial is a step-by-step introduction to rudimentary knowledge of Query 3 utilization. The advanced tutorial presents data base maintenance commands, alteration of human interface, and graphics features. Currently little centrally located or on-line documentation exists to assist individuals in gaining familiarity with Query 3. This work is intended to fill this documentation gap. It is hoped that it will prove useful to students and other individuals interested in becoming familiar with Query 3 as an area of command and control research.

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

QUERY 3; A DATA BASE INQUIRY SYSTEM DESCRIPTION AND USER'S TUTORIALS

by

Leo Joseph Faneuf III
Lieutenant Commander, United States Navy
B.S., United States Naval Academy 1969

Submitted In Partial Fulfillment Of The
Requirement For The Degree Of

MASTER OF SCIENCE IN SYSTEMS TECHNOLOGY (C3)

from the

NAVAL POSTGRADUATE SCHOOL
March 1980

ABSTRACT

This work examines a data base inquiry system called Query 3. A demonstration version of this experimental system is available on the Advanced Research Projects Agency Network (ARPANET) at Computer Host 116.

This research contains a description of the system, and basic and advanced tutorial user's guides. System description covers the various aspects of Query 3 including it's online system (NLS) base, ARPANET operations, human interface capabilities, system operation, and data structure. The basic tutorial is a step-by-step introduction to rudimentary knowledge of Query 3 utilization. The advanced tutorial presents data base maintenance commands, alteration of human interface, and graphics features.

Currently little centrally located or on-line documentation exists to assist individuals in gaining familiarity with Query 3. This work is intended to fill this documentation gap. It is hoped that it will prove useful to students and other individuals interested in becoming familiar with Query 3 as an area of command and control research.

TABLE OF CONTENTS

LIST OF FIGURES

Figures

I. INTRODUCTION

The evolution of Military Command and Control (C2) Systems has caused great emphasis to be placed on computer utilization to provide increased C2 capabilities. Computers have been used effectively in such C2 related areas as communications, photo analysis, weapons control/guidance systems, and tactical data systems. Additionally, a great amount of effort has been expended in the critical fields of data base management and information systems.

Data base management/information system research and development efforts have lead to better human interface with the computer. Many experimental systems use English-like commands or phrases to access stored information and to carry out operations. These types of systems may well be acquired for military use in the future.

Advances in inter-computer communications through packet switching techniques and network structuring have made the concept of distributed data bases a reality. These data bases are capable of exchanging and updating information quickly and efficiently. Distributed data base systems provide many military benefits including greater flexibility and increased survivability through redundancy. The Advanced Research Projects Agency Network (ARPANET), an experimental network under the operational cognizance of the Defense Communications Agency, is an example of a functioning computerized, packet switching network.

The Query 3 system, with which this thesis is concerned, was developed under a Navy contract and is available on the ARPANET to demonstrate a data base management and inquiry system in a Navy command and control environment. The Query 3 system has been through several iterations and allows the user to access a structured data base in a manner that approaches the natural use of English. Additionally, the fact that Query 3 resides within the ARPANET provides the user with the opportunity to observe the advantages of modern inter-computer communications.

The purpose of this research was to discuss the mechanics of the Query 3 system through exposure to Online System (NLS) and a description of Query 3 system development. Additionally, a User's Guide to the Query 3 system was developed to present a non-technically oriented user with an easy method of mastering the system.

II. BACKGROUND

A. DATA BASE MANAGEMENT SYSTEMS

A data base can be defined as an orderly collection of facts stored to serve information requirements. The ideal data base would contain all information required in the process of conducting one's business and would contain current, historical, and forecast information which would be stored logically and be available for quick retrieval. [4] In today's world, the creation and management of computer data base systems which attempt to encompass the foregoing requirements is a reality, and has lead to what has been termed "an information explosion."

Most data base management systems have evolved from simple file-accessing aids and retrieval packages. This evolution has paralleled the advances in computer technology. Data base management systems are designed to store, structure, and manipulate the data in a manner that is separate from and "invisible" to the user or using programs. [9] Data base management systems provide an interface between the user and the data base and are composed of sets of computer programs that translate user commands into actions performed on the data base. See Figure 1. [8]

A data base management system performs three functions: translate, transform, and transfer. The system translates the user's requests to determine the desired data base operations.

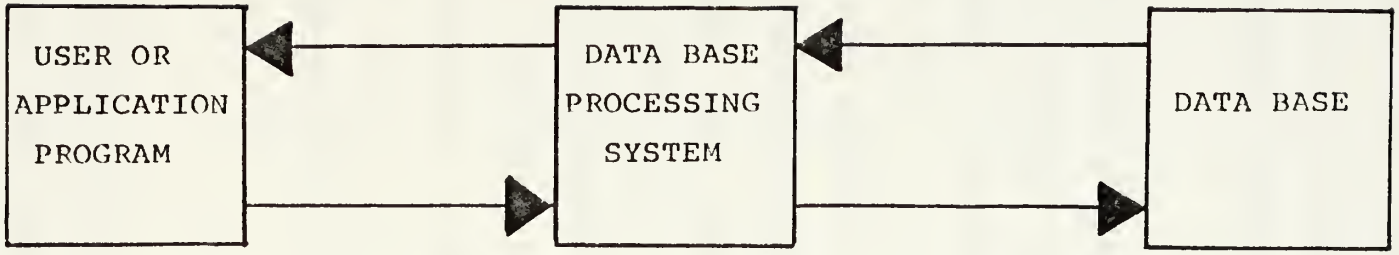


FIGURE 1. Data Base Processing Components
Adapted From Reference [8]

Requests specified by logical relationships are transformed into operations to be performed on the data base. The data is then transferred between main memory and secondary storage. [8]

If interactive processing is used, there must be a communications processor to provide synchronization at control of the terminals in the system. The processor interacts with the data base, which in turn interacts with the host operating system. The host operating system performs job and data management and completes the physical input and output to the data base files. See Figure 2. [8]

A bit, or binary digit, is the basic building block of computer stored information. Bits are grouped into bytes (characters), bytes are grouped into fields, and fields are grouped into records. A collection of records is commonly called a file. Files can be processed either sequentially or randomly. Some methods of file structure are shown in Figure 3. [9]

B. QUERY 3

The development of the Query family of subsystems was initiated by the Naval Electronics Laboratory Command (now called Naval Ocean Systems Center) as a part of the capabilities assessment task of its Project 0732600. This task involved an investigation of the potential of the Stanford Research Institute (SRI) Online System (NLS) for applicability to Navy command and control problems. The initial effort was

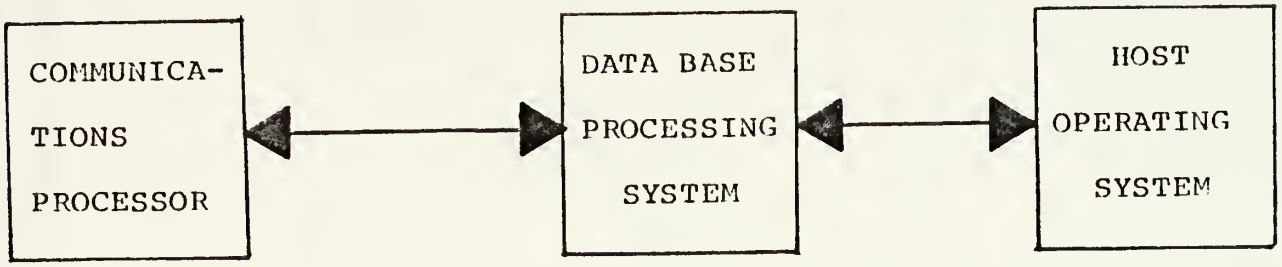
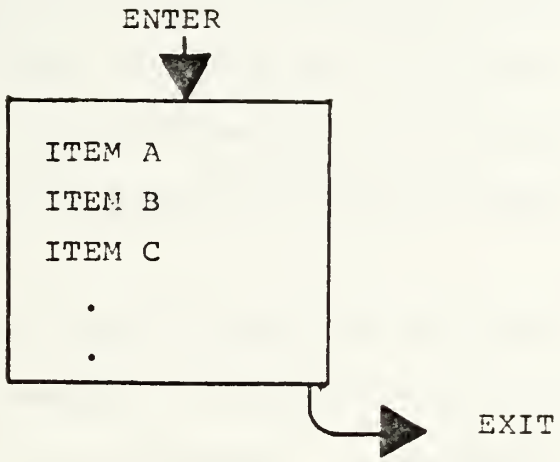
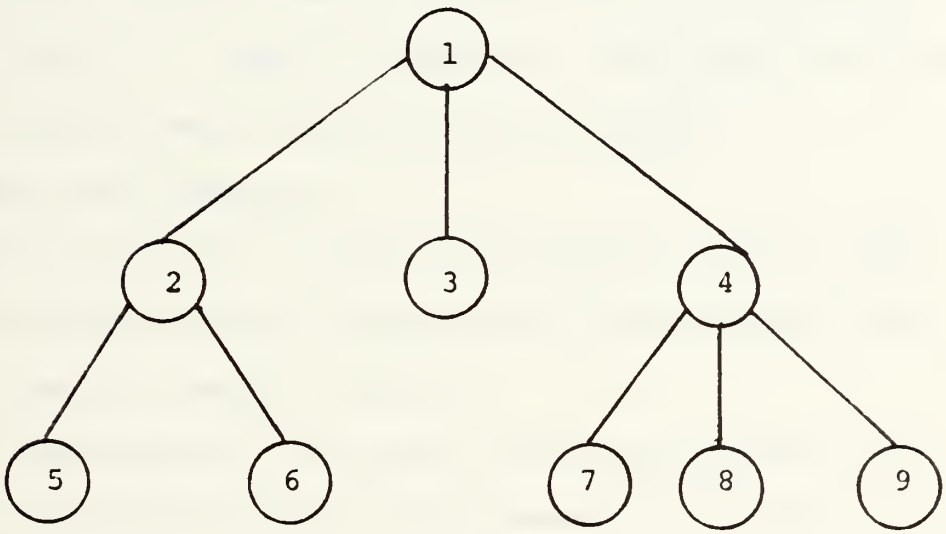


FIGURE 2. Interactive Processing
Adapted From Reference [8]

A. LISTS



B. TREES



C. STRINGS

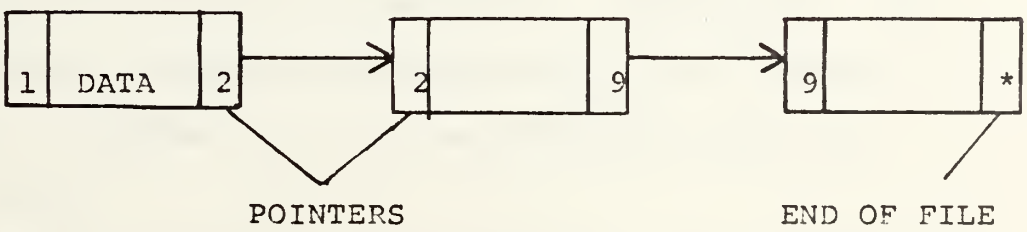


FIGURE 3. File Structures

called Query and was developed by SRI in accordance with specifications supplied by NELC. Query was an NLS-based data management system that was used to interrogate and manipulate a relatively small and simple facsimile of a Navy command and control data base. [10]

The initial Query version took two and one-half days to design and program. It possessed a static data base of 70 ships and supported two data base commands. Improvements to Query were made in follow on versions of the system called Query 2 and Query 3. Query 2 required a month and a half to program and incorporated the following changes:

-- 15 data base commands
-- A dynamic data base of 250 ships/planes and 110 ports
-- Capability of moving the ships and planes within the data base on command. [10]

Query 3 represented a six month programming effort and provided an order of magnitude improvement over Query 2. Query 3 had over 30 data base commands and increased dynamic data base of 500 ships/aircraft and 130 ports. The data base was spread over two oceans (Atlantic and Pacific) which offered further realism and complexity. Additionally, previously existing commands were enhanced to allow the user to manipulate the system in a more operationally realistic manner. [10]

The Query 3 system was designed to be a self-documenting demonstration of the use of a structured data base query system. Inquiry and data retrieval is accomplished through

the use of phrases that approach the natural utilization of English. Additional structure is provided through the use of prompting so that the user will never be uncertain as to what input is required. [12]

III. ONLINE SYSTEM (NLS)

NLS, in which Query 3 is written, is a sophisticated computer software system designed to assist users in working with information. NLS provides users with a rather wide range of information management tools, from a simple set of commands for reading, writing, and printing documents and drawings to sophisticated methods for retrieving and communicating information. NLS was originally developed at Stanford Research Institute (SRI), but is now supported and under continuing development by Tymshare Corporation. [2]

NLS forms the basis for the Query 3 system. All of the data base management and inquiry features of Query 3 are a direct result of the capabilities inherent in NLS. In order to understand how Query 3 functions, one must first have an exposure to NLS. The remainder of this section will present an overview of NLS to assist in understanding follow-on sections describing Query 3.

A. SUBSYSTEMS

NLS is divided into subsystems which are subordinate systems that make up the whole. NLS subsystems consist of sets of commands that are related to particular activities. When entering NLS the user is normally put into the "base" subsystem. The base subsystem is the major subsystem of NLS. It has commands that allow the user to name, read, and write information online and output it to hard copy. [2]

Other NLS subsystems and their descriptions are as follows:

-- Calculator: Allows the user to do arithmetic in a way that is integrated with the rest of NLS.
-- Programs: Constitutes user programming software by which the user may write special functions or load a program or subsystem that is already written and compiled.
-- Send Mail: Allows the user to send messages and documents to a list of people known to NLS and have these messages cataloged and stored.
-- TENEX Time Sharing System: This Time Sharing System actually supports NLS which runs as a subsystem of TENEX. NLS draws heavily on TENEX for file handling. TENEX may be called as a subsystem of NLS via an appropriate command. TENEX will be discussed more completely in a later section. (Note - the TOPS 20 System is also utilized on the ARPANET and is capable of supporting NLS).
-- User Options: Allows the user to alter how he interacts with NLS to fit his own equipment, use patterns and style by specification of parameters. [1]

NLS permits the user to enter and use various subsystems as desired. Access is normally accomplished from the program subsystem. Each subsystem also prints identifying character(s) at the left hand margin of the user's terminal screen or teletype paper. These characters are called "heralds" and are usually comprised of the first few characters of the subsystem's name. [2]

B. COMMANDS

It was previously indicated that NLS subsystems are composed of sets of commands. Commands are essentially instructions given to the computer to perform an action. NLS commands are normally inputed to the computer as words which consist of a set of characters (letters, numbers, punctuation marks, etc.). NLS is programmed to recognize these command words and takes the specified action in response to correctly entered command words.

The following examples of commands normally encountered in the base subsystem are presented to assist in understanding the concept:

Create File: This command makes a new file in the user's directory.

Delete: This command is used to remove information, such as a character, a word, or some text.

Insert: This command is used to add new information to a file.

Jump Link: This command is used to move from one file to another.

Replace: This command is used to remove a character, word or some text, and put new information in it's place. [17]

The previous examples illustrate that commands in NLS are relatively simple for the user to understand since the command words are designed to approximate their conventional meanings in normal use of the English language. The mechanism that allows the user to perform interaction with the computer

through the use of command words is called the Command Meta Language or CML.

Command Meta Language is a formal language developed at SRI's Augmentation Research Center (ARC) that describes the command language and interaction of the subsystem with its human user. CML works in conjunction with L10 (Language 10), an Algol-like procedure oriented programming language. CML constitutes the front end of the NLS while L10 is used in the supporting programs, or back end of the system. [1]

C. TENEX TIME SHARING SYSTEM

The TENEX Time Sharing System supports NLS on the PDP-10 computer system. NLS runs as a subsystem of TENEX (which is called the "superior executive" at this top level) and draws extensively on TENEX's file handling capabilities. TENEX is also available as a subsystem of NLS via the "goto TENEX" command. Figure 4 illustrates the relationship of the TENEX system, CML, L10, and NLS. [1]

In addition to providing the overall operating environment for NLS, the TENEX Time Sharing System has several aspects which are of interest to the user and facilitate the use of NLS. The first is that TENEX contains various susbystems, such as NLS. These subsystems can perform specific functions (i.e., text editing) and can be used together with other subsystems in certain circumstances. Other TENEX subsystems are entirely self-contained and require little knowledge of the overall TENEX system or other subsystems. [5]

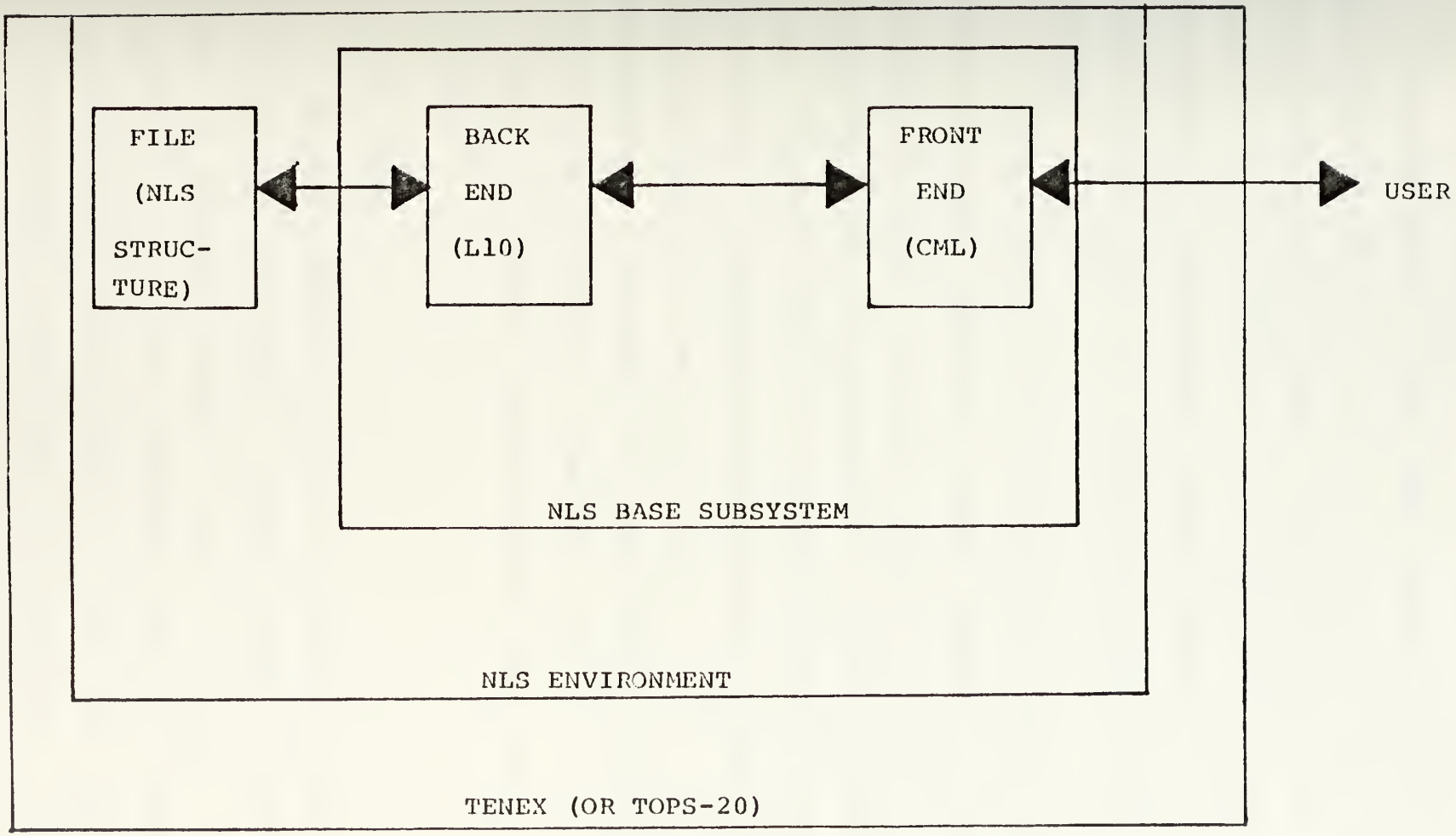


FIGURE 4. TENEX/CML/L10/NLS RELATIONSHIPS

A second aspect of this TENEX system is its file system. The TENEX file system is a vital portion of NLS. The TENEX file system provides access to and control over the various input and output devices in the system. Paper tape readers, line printers and terminals can be treated and accessed as files. TENEX files can be stored on magnetic tape, disk, or DEC tape. /-5-/

A final TENEX capability is called a virtual computer. The virtual computer gives the user the ability to run machine language programs. The term "computer" is used because the capability gives the appearance of a piece of computing hardware. The term "virtual" is used because most of the capability is provided by the system software that controls TENEX. The virtual computer is important because it allows the use of a more powerful core memory and input-output system than the system hardware actually possesses. /-5-/

NOTE: The TOPS-20 System provides the same type of support as TENEX.

D. NLS FILES

A primary purpose of NLS is to permit the user to write documents and store them in computer files for further action. Files are the basic units in a user's directory. NLS files are composed of one or more statements that can be arranged at different levels in an outline or "tree" form. /-I-/ For example, the first chapter title would be at position 1 in the outline, the first subheading in that chapter would be at position 1A, and the first paragraph following that subheading would be at

position 1A1. [2] This structure is called "hierarchy" and the interrelations between statements are called structural relationships. [1]

The basic component of the NLS file structure is the statement. The statement is composed of strings consisting of 1 to 2000 characters. A statement may be a character, line, sentence, heading, paragarph or graphic. A statement is given a statement name and a statement number. Statement names are a string of characters and "name" a statement so that the user can point to the statement by using its name as an address. A statement number indicates the exact position of the statement within the file structure. Statement numbers are not permanent parts of the statement and change as the position of the statement changes. [1]

The ability to locate statements in an NLS file is important to the user. NLS files are stored in a random order. A system of pointers is used to retrieve statements.

NLS provides many commands that deal with whole files allowing the user to make changes, erase changes made, send files to people, delete files, transfer files from one directory site to another, and return to recent files the user has accessed. Editing commands permit the user to control logical sections of a structured file and to display the overall picture. The user has the power to revise or reorganize the file as desired. It is just as easy to delete, move, replace, insert, copy, and transfer statements, branches and groups as it is to control characters, words and text. [2]

The hierarchial structure of NLS files provides the user with several advantages while looking at online information. By using various viewing specifications (view specs), one can view the information in many different ways. When reading a document in a file, the user can skip from one section in the document to another according to their logical positions within the hierarchy. Commands allow the user to go to a specific location within a document or he can scan the file until the appropriate section is found. [2]

E. SUMMARY

NLS provides a coherent environment from which various cooperating subsystems can be accessed and utilized. The main features of NLS that enable the user to operate in that environment are:

- -- NLS supports interactive devices (i.e., teleprinter and two dimensional cathode ray tube).
- -- NLS interprets and passes user commands to the appropriate subsystems.
- -- NLS maintains a hierarchically structured file system.
- -- NLS commands permit editing, manipulation, cross-referencing, and cross copying hierarchically arranged blocks of mixed text and graphics.
- -- NLS formats and outputs information obtained from files and other input sources to hardcopy or microfilm.
- -- NLS provides a wide range of tools to aid applications in data management, document production, message handling, software engineering, and management. [10]

IV. DESCRIPTION OF QUERY 3

A. HOW QUERY 3 FUNCTIONS

Query 3 is an NLS-based data management system that can be used to interrogate and manipulate a reasonable facsimile of a Navy command and control data base. The system design is such that the user is given a robust taste of the NLS environment utilized in an operationally meaningful way. Query 3 possesses the additional capability of allowing the user to interface with the system utilizing commands that approach the natural use of the English language. This interface capability is further enhanced through structuring which ensures that the user will never be uncertain of what input the system requires next. [10]

1. Query 3 Hardware

A Query 3 work station would normally contain the following hardware:

- Standard key board with cathode ray tube (CRT).
- Line processor.
- Graphics display terminal.
- Host computer (either present or accessed via a network; normally a PDP-10 computer).

The specific CRT utilized with the system in a graphical mode is optional subject to the limitation that the CRT must be a video terminal which is a stand-alone, separable terminal containing alpha-numeric display capability, a key board,

storage, control logic and an asynchronous communications interface. When the graphical capability is not being utilized a teletype like terminal, such as the Computer Devices Inc. Miniterm, may be used in place of the CRT if desired. [5]

The line processor is a device that is used in support of the graphics display terminal. A TEKTRONIX 4014 storage tube graphics display system may be used to provide graphical display for Query 3. The 4014 contains a 19-inch flicker free storage tube, a standard ASCII keyboard, thumbwheels, a joystick. It's modes of operation include alpha-numeric, vector, and dashed or dotted vector. [5]

2. Query 3 Software

Query 3 utilizes essentially the same software as used in NLS. Some augmentation to the NLS software was required and will be discussed later in this chapter. A system diagram is provided in Figure 5.

Query 3 exists in the TENEX (or TOPS 20) environment which may be accessed on the ARPANET. The user communicates with the Query 3 front end which interfaces with the Query 3 back end and an NLS data base. The front end may also be interfaced with the NLS graphics subsystem and a data computer which can support large data bases. [10]

3. Query 3 Commands

When performing operations in the Query 3 system, the user is able to interface with the system by using various command words. These commands are similar to those used in NLS and permit the user to either inquire about specific

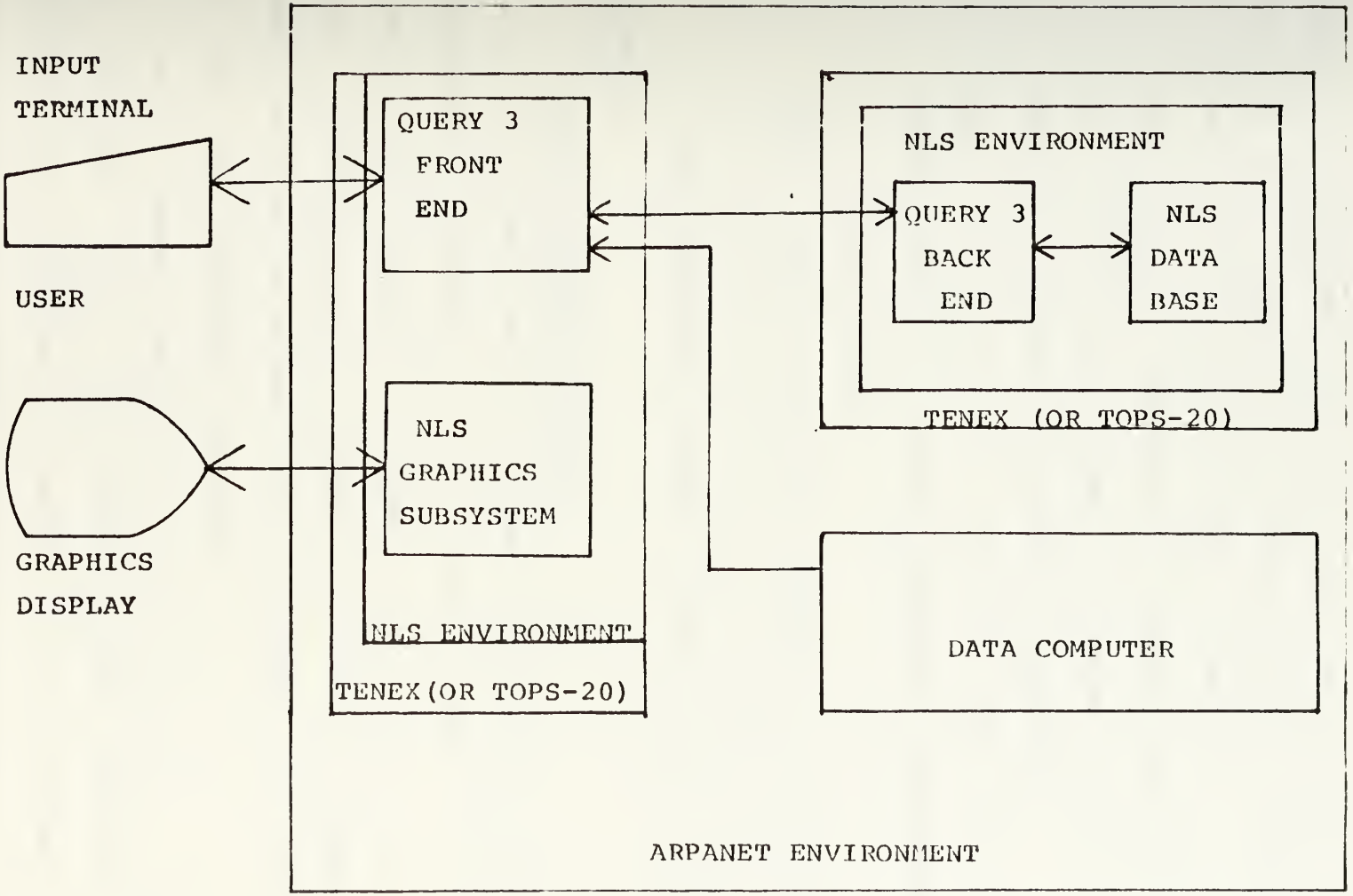


FIGURE 5. Query 3 Diagram

information contained within the data base or cause a general alteration of the position of platforms within the data base. The following are examples of Query 3 commands:

How (far is) - Determine the distance and bearing of one platform with respect to another or the time that would be required for them to rendezvous.

Move - Move all craft in the data base for a specified number of hours with the appropriate percentages of fuel being consumed.

Show - The most versatile command. It allows the user to interrogate the data base on various fields. [12]

A complete list of Query 3 commands may be found in Appendix A.

4. Query 3 Human Interface

The primary method of human interface with Query 3 is through the key board of a CRT. Once the user is on the system, a herald, the name Query 3, appears on the left side of the screen indicating system readiness to accept a command. A command word may be entered into the system. The subsystem will make the appropriate response.

Query 3 commands demonstrate the flexibility of the NLS front end. [12] The method of human interface can be modified. For instance, the herald can be shortened or eliminated totally. The command recognition mode can be altered to allow the user to type either the exact number of characters required for disambiguation or the first three characters of each command word before the command is recognized. In the

normal mode (called terse) the command "PUT" for example, is recognized after the first letter "P" is typed. In the verbose mode the "PUT" command is recognized only after the full word "PUT" is typed. Similarly, the command "SHOW" is recognized by typing the letter "S" when in the terse mode but is recognized only after typing the entire word show when in the verbose mode. [10]

5. Query 3 Graphics

The NLS foundation upon which Query 3 is constructed contains a comprehensive software package for generating and viewing line drawings consisting of mixed text and graphics. This NLS graphics system is interfaced to Query 3 by means of an option in both the "FIND" and "SHOW" commands that allow the user to obtain a graphical representation of the selected platforms in place of the normal textual display. This Query 3 facility is accessible only from terminals equipped with the necessary graphical display devices. [10]

Platforms are mapped onto the display screen using mercator projection. The portion of the sphere displayed and, therefore the magnification, is selected automatically by Query 3, rather than being set by the user. Each platform is represented as a small circle, square, diamond or triangle and labeled with its name or class. [10]

B. SYSTEM OVERVIEW

1. Introduction

The Query 3 subsystem is available for use on the ARPANET. The ARPANET is a network that links a number

of computers located at various research centers in Europe and the United States. The capabilities inherent in the ARPANET represent the many advances in computer technology that make the use of distributed data bases feasible and thereby enhance subsystems such as Query 3.

A demonstration version of Query 3, located on ARPANET Host Computer 116, was utilized for this research. Therefore, the following discussions are provided to give an appreciation of the ARPANET-QUERY3 relationship.

2. The ARPANET

a. What Is The ARPANET

The ARPANET is an operational, resource sharing, host-to-host network linking a wide variety of computers at research centers sponsored by the Defense Advanced Research Projects Agency (DARPA) and other DOD and non-DOD activities in the continental United States, Hawaii, Norway and England.

The ARPANET originated as a purely experimental network in late 1969 under a research and development program sponsored by DARPA to advance the state-of-the-art in computer internetting. The network was designed to provide efficient communications between heterogeneous computers so that hardware, software, and data resources could be conveniently and economically shared by a wide community of users. As the network successfully attained its initial design goals, additional users were authorized access to the network. Today the ARPANET provides support for a large number of DOD and non-DOD government projects with an operational network of many nodes and host computers.

Following the successful accomplishment of initial ARPANET design goals and the expansion of the network, it was considered appropriate to transfer the responsibility for operation of the ARPANET from DARPA to the Defense Communications Agency (DCA). In July 1975 the DCA became the operational manager of the ARPANET. [3]

b. A Brief Description Of The ARPANET

The ARPANET is an operational, computerized packet switching DOD digital network which provides a capability for terminals or geographically separated computers, called hosts, to communicate with each other. The host computers differ from one another in type, speed, word length, operating system, and other characteristics. Each terminal or host computer is connected into the network through a small local node computer called an IMP or TIP. The computer network is formed by interconnecting the IMPs through wideband communications lines (maximum 50,000 bits per second) supplied by common carriers. Figure 6 is a geographic picture of the ARPANET.

Each node is programmed to receive and forward information packages to the neighboring nodes in the network. During a typical operation, a host passes an information package to its node; the information is passed from node to node through the network until it finally arrives at the destination IMP which in turn passes the information package to the destination host. This process normally takes less than 250 milliseconds.

ARPANET GEOGRAPHIC MAP, JANUARY 1979

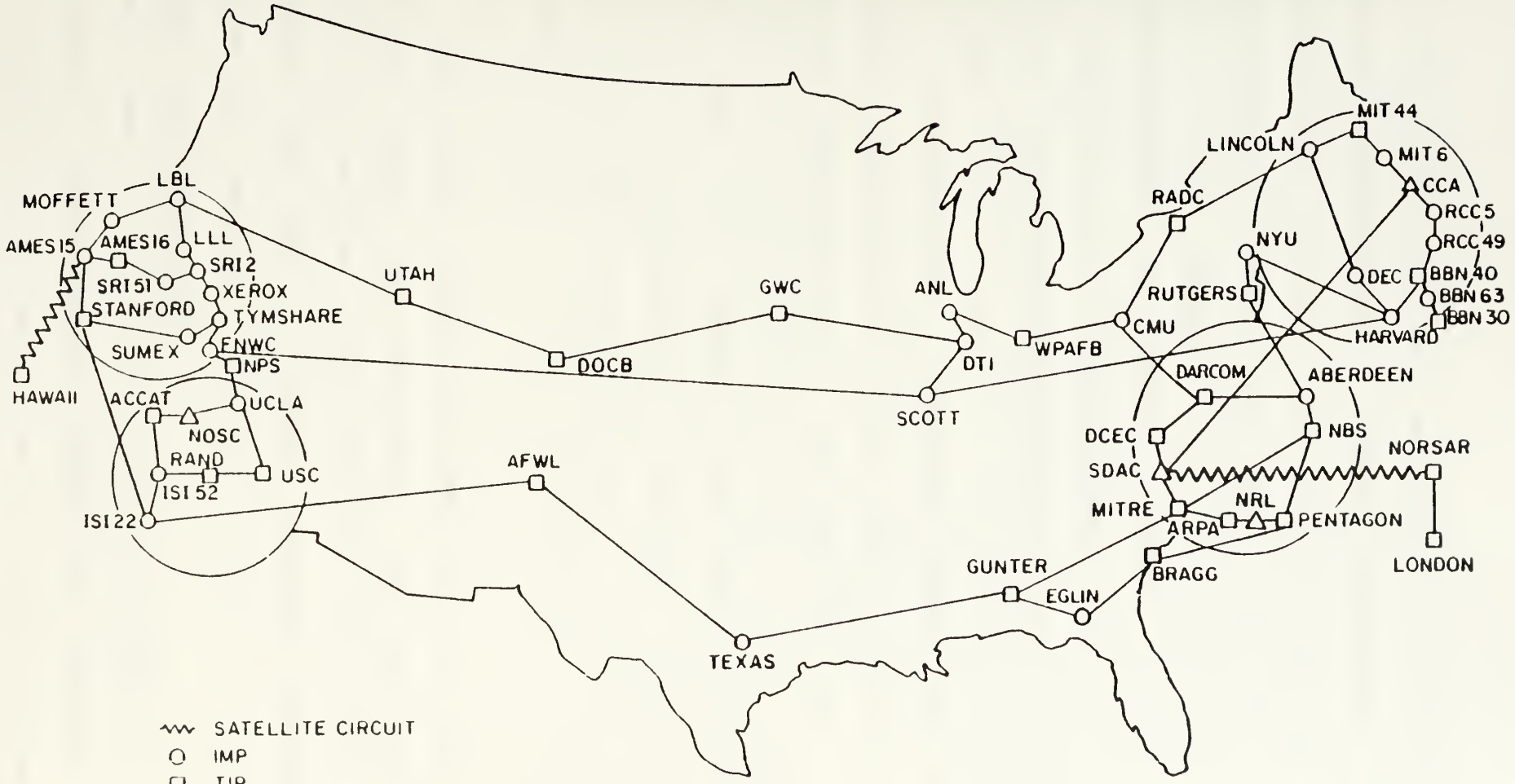


FIGURE 6. ARPANET Geographic Map
Adapted From Reference [3]

Hosts communicate with each other via information packages called regular messages. A regular message may vary in length from .96 to 8159 bits, the first 16 of which are control bits called the leader. The leader is also used for sending control messages between the host and its IMP or TIP (node). The remainder of the message is the data or text.

After receiving a regular message from a host connected to it, a node breaks the message into several packets and passes these through the network in the direction of the destination. Eventually, when all packets arrive at the destination, they are reassembled to form the original message which is passed to the destination host. The destination node returns a positive acknowledgement for receipt of the message to the source host. If the message is not received due to a nodal failure or line disruption, an incomplete transmission message will be returned to the source host. Procedures also exist to ensure receipt of positive acknowledgement or incomplete messages.

Users of the ARPANET may access local or distant server computers (hosts) over the network. They may also exchange information packages, create realtime links between users, transfer files from one computer to another, and submit batch jobs to other computers. [3]

3. Query 3 And The ARPANET

Query 3 may be accessed by entering the TENEX (or TOPS 20) system via the appropriate ARPANET Host Computer [for the purpose of this research Host 116 (ISIE) contained

the Query 3 demonstration [7]. When the user has been accepted by the host computer, he must then log in to the directory containing the Query 3 system. The directory available for this research was named Query 3.

After entering the appropriate directory, the user enters NLS "BASE" subsystem. At this point the "PROGRAM" subsystem is called and the Query 3 program is loaded. Query 3 is then operated as a subsystem of NLS and, like NLS, Query 3 is capable of using its superior systems (i.e., NLS and TENEX) as subsystems. Figure 7 provides a diagram of the forgoing discussion.

C. DATA BASE STRUCTURE AND CONTENT

The Query 3 data base contains a two ocean data base with over 75 ports and 250 craft in each ocean. Individual files for craft and ports contain information that may be static or dynamic in nature. This information is entered and presented using U.S. Naval terminology. Therefore, the following definitions are presented to assist those unfamiliar with Naval terminology:

Platform: A platform can be either a port, an aircraft, a surface vessel, or a submarine.

Category: A functional classification assigned to a platform to indicate its use; for example NAV for naval, MER for merchant.

Type: A functional category of craft, such as aircraft carriers, submarines, etc.

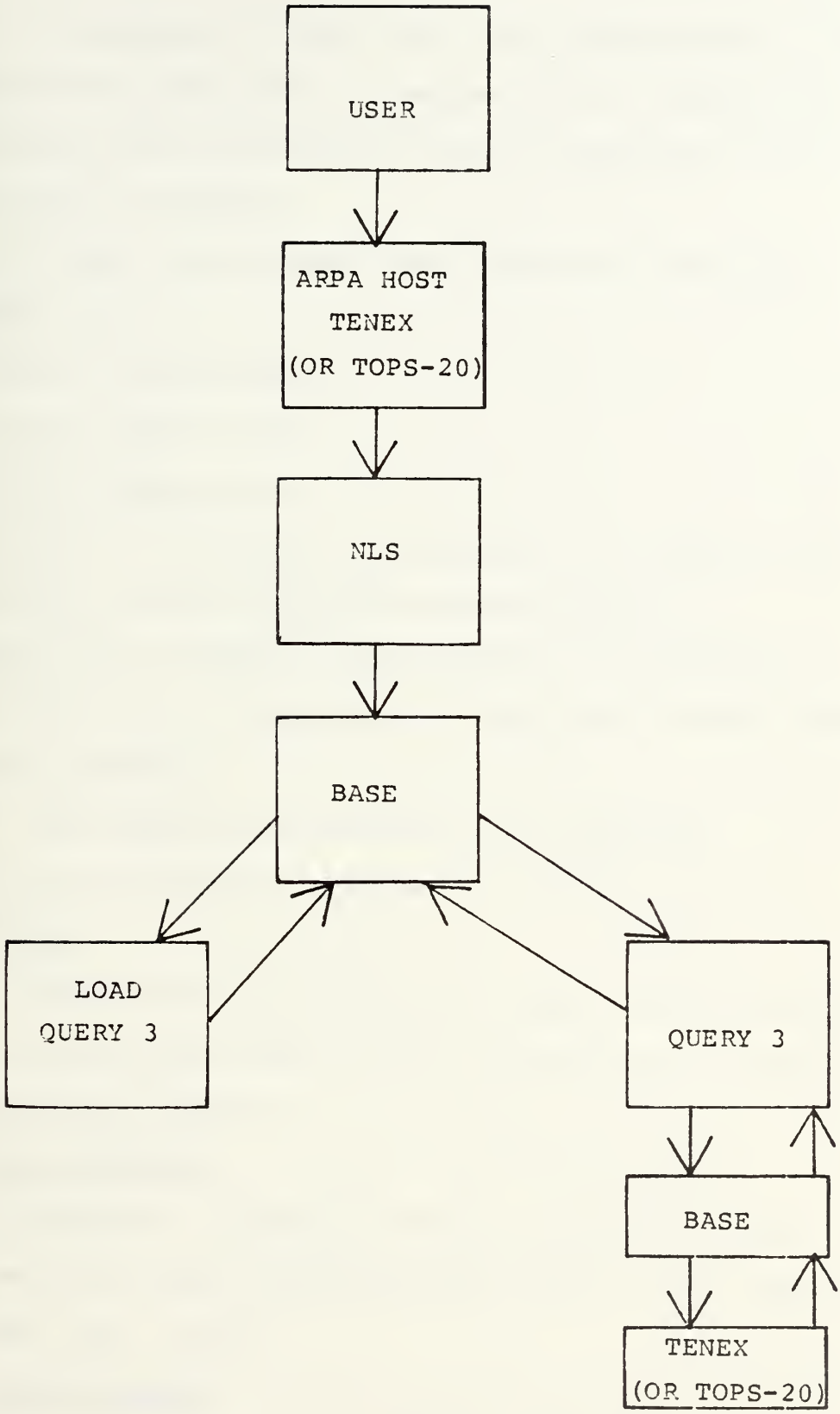


FIGURE 7. Query 3 And The ARPANET
Adapted From Reference [11]

Class: A category of craft that were constructed to almost identical specifications. Examples are the Belknap class of cruisers, the Los Angeles class of submarines, and the Knox class of frigates.

Country Code: A two letter code denoting a particular country, e.g.,

US for United States

UR for Soviet Union

JP for Japan, etc.

Task Force: A collection of geographically close craft that travel together to perform some mission.

Latitude and Longitude: The geographical coordinates of a platform given in alphanumeric form such as 4500N 15043W, or 45-00N 137-45E.

OPCON: The operational control of the platform.

Course: The direction in degrees in which a craft is travelling.

Range: Defined for the purposes of Query 3 as the distance that a platform can travel with a 100 percent fuel load at maximum speed. [12]

1. Static Records

In the Query 3 subsystem static records are ports. These records have only static characteristics such as name, position, country code, etc. [11]

2. Dynamic Records

Dynamic records are the various craft located within the data base, i.e., ships and aircraft. These records contain

both static and dynamic characteristics. Examples of static characteristics are name, country code, category and type. Examples of dynamic characteristics are position, speed, and bearing. [11]

Data base dynamic records are capable of being updated as the situation requires. Samples of dynamic records follow and are indicative of how the data base records are maintained:

a. Hostile Record

(ADMIRAL FOKIN)L=KYNDA:N=UR:U=NAV:V=S:T=CLG:P=7211N125E:B=106:S=15.0:F=59.10:D=Y:

7243N009W	1440 9DEC77	A	GOOD FIX
7239N039E	024010DEC77	A	
7226N104E	144010DEC77	D	POOR FIX
7213N037E	024011DEC77	A	GOOD FIX
7211N125E	144011DEC77	E	GOOD FIX

b. Merchant Record

(ADELAIDE@STAR)L=BLUESTAR:N=UK:U=MER:V=S:T=BULK:P=5744N747E:S=18.0:B=286:F=94.85:D=N:Y=WHEAT:Z=150T:

5712N612E	1440 9DEC77	E	
5730N615E	024010DEC77	B	POOR FIX
5741N641E	144010DEC77	A	
5744N714E	024011DEC77	B	POOR FIX
5744N747E	144011DEC77	D	POOR FIX

c. Friendly Record

(AINSWORTH)L=KNOX:N=US:U=NAV:V=S:T=FF:P=3739N2258E:S=21:B=155:F=37.00:D=N:0=TG67.1:C=CDR D. CHAROLLETTE:H=1090:

3738N2351E	1440 9DEC77	A	GOOD FIX
3746N2326E	024010DEC77	F	POOR FIX
3731N2307E	144010DEC77	C	
3748N2322E	024011DEC77	A	
3739N2258E	144011DEC77	C	

The static portions of the records are items such as name (Ainsworth), class (L=Knox) and country code (N=US). The dynamic portions of the records are items such

as position (P=3739N2258E) and OPCON (0=TG67.1). In addition the data base maintains a record of reports on the craft's postions as received with an indication of the report's reliability (i.e., Good Fix, Poor Fix). [11]

3. Sets Of Records

The Query 3 data base also contains items that are termed sets of records. These data items are used to fill the data fields of static/dynamic records and consist of those data items that are common to many of the port/craft records. Pointers are used to include the appropriate data items in port/craft records thereby reducing redundant entries in the data base. Figure 8 is a listing of sets of records.

A comparison of the "AINSWORTH" example of a dynamic record with Figure 8 shows that the fields L=KNOX, N=US, U=NAV, V=S, T=FF, D=N, and 0=TG67.1 were filled from the record sets Class, Flag, CAT, Plat, Type, Doctor, and OPCON. [11]

4. Data Computer

In previous versions of Query the NLS file system was utilized as the basis for the data base management system. This yielded satisfactory results with the smaller data bases associated with those versions but was considered ill-suited to the data base requirements of a realistic command and control application. Therefore, Query 3 was designed to use as its data base either an NLS file or a data base maintained on a data computer.

The data computer is accessed via the ARPANET by means of network connection and data language. To minimize

```
(SETS) OF RECORDS
   (CARGOTYPE)
   (ACFT)
   (CHROME)
   (COAL)
   (CONST)
   (FARMAC)
   (FOOD)
   (GENMER)
   (OIL)
   (TANKS)
   (TIN)
   (TRUCK)
   (TUNGST)
   (UNAD)
   (WHEAT)
(CAT)
   (COM)
   (MER)
   (NAV)
(CLASS)
   (42)
   (47)
   (ADAMS)
   (ALINDA)
   (AMSTERDAM)
   (AUDAC)
   (BEAR)
   (BELKNAP)
   (BISON)
   (BLACK@SWAN)
   (BLUESTAR)
   (BOEING747)
   (CHARLIE)
   (CONCORDE)
   (DELTA)
   (E-4)
   (ECHO@II)
   (ENDEAVOUR)
   (FORRESTAL)
   (HASSAYAMPA)
   (HUNT)
   (KASHIN)
   (KAZBEK)
   (KITTYHAWK)
   (KNOX)
   (KONLN)
   (KRESTA@II)
   (KURIL)
   (KYNDA)
   (LEAHY)
   (LEANDER)
   (LOS@ANGELES)
   (LUTJENS)
   (MACKENZIE)
   (MAY)
   (MISSION)
   (MOSKVA)
   (NIARCHOS)
   (NIMITZ)
   (OKEAN)
   (ORION)
   (RIVER)
   (SEALIFT)
   (SPRINGBOK)
   (STINNES)
   (STURGEON)
   (SVERDLOV)
   (UNNAMED)
   (VICTORY)
   (VIRGINIA)
   (WILHELMSON)
   (WILSA)
   (YANKEE)
(DOCTR)
   (N)
   (Y)
(FLAG)
   (AL)
   (AN)
   (AR)
   (BE)
   (BR)
   (CA)
   (CI)
   (CL)
   (CU)
   (CY)
   (DE)
   (EG)
   (FI)
   (FR)
   (GR)
   (IT)
   (JA)
   (LA)
   (LI)
   (MX)
   (NE)
   (NO)
   (PA)
   (PL)
   (PO)
   (SA)
   (SF)
   (SP)
   (TU)
   (UK)
   (UR)
   (US)
   (VE)
   (WG)
   (LABELS)
(MISCELLANEIOR)
   (CLASSES)
   (CRAFT)
   (PLATFORMS)
   (UNK)
(OPCON)
   (CTE22:2:2.1)
   (CTF21)
   (CTF22)
   (CTF27)
   (CTF67)
   (CTG21)
   (CTG21.4)
   (CTG22.2)
   (CTG22.3)
   (CTG27.6)
   (CTG27.7)
   (CTG67.1)
   (CTG67.2)
   (CTG67.4)
   (CTU21.7.2)
   (CTU22.2.1)
   (CTU22.3.1)
   (CTU22.3.2)
   (CTU27.7.2)
   (TF29)
   (TF65)
   (TG21.2)
   (TG21.4)
   (TG22.2)
   (TG22.3)
   (TG27.6)
   (TG27.7)
   (TG67.1)
   (TG67.2)
   (TU21.1.3)
   (TU21.2.3)
   (TU21.7.2)
   (TU22.2.1)
   (TU22.3.1)
   (TU22.3.2)
   (TU27.7.2)
   (TU65.2.1)
   (TU67.4.1)
   (TU67.4.2)
)PLAT
   (A)
   (P)
   (S)
   (U)
(SYNONYMS)
   (BROWN)
   (BYRD)
   (DANIELS)
   (HART)
   (IKE)
   (JFK)
   (RIVERS)
   (STANDLEY)
   (STODDART)
   (SUBLANT)
   (TURNER)
   (YARNELL)
(TYPE)
   (AGI)
   (ANCPI)
   (AO)
   (BEAR)
   (BISON)
   (BULK)
   (CA)
   (CLG)
   (CLGN)
   (COM)
   (CV)
   (CVA)
   (CVAN)
   (DDG)
   (FF)
   (FFG)
   (MAY)
   (ORION)
   (SSBN)
   (SSGN)
   (SSN)
   TNKR)
```

FIGURE 8. Sets Of Records

the number of interhost exchanges when using the data computer, Query 3 was designed to maintain the results of the last show or find command locally in an NLS file. Subsequent questions are then answered without accessing the data computer. [10]

5. Data Base Efficiency

The most important characteristic of previous Query demonstrations was that user interface, as specified in SRI's Command Meta Language, could be rapidly changed. Query and Query 2 were not particularly effective data management systems since they were built upon the NLS file system which was designed primarily for other applications. [10]

Data base management inefficiencies detracted from the system operation. Therefore, Query 3 was designed to search it's data base more efficiently. Specifically, the Query 3 data base was inverted on platform names to reduce the time required to locate platforms by name. Also the mechanism by which fields are extracted from a record were improved so that a single record is scanned only once, regardless of the number of fields required or the number of times those fields must be examined. These features resulted in decreasing the execution time of Query 3 commands. [10]

D. OPERATION UPON DATA STRUCTURE

Query 3 provides the user with the capability to extract information from the data base through the use of commands. The commands allow the user to specify the criteria desired when selecting information for retrieval. For example, the find command permits the use of any of the following criteria

when selecting a platform(s) for display:

-- Containing a constant, for the composite data base field synonyms.
-- Equal to a constant, for the textual fields name, flag, position, platform, category, type, class, CO, cargo quantity, and OPCON.
-- Less than, equal to, or greater than a constant for the numeric fields bearing, hours, range, speed, maximum speed, and percent fuel.
-- True, for the Boolean Field Doctor.
-- Known or defined, for any field. /~10_7

A sample find command illustrating the above follows:

Query 3 C: FIND(ALL) C: CRAFT (WITH)C:
PERCENT (FUEL LEFT) C: LESS(THAN) T/ /~A7:95

The reader should note that this command example utilizes the numeric constant 95 as the criterion for finding those craft with less than that percentage of fuel. The Query 3 subsystem will search the data base for craft meeting the criterion and then provide the information in a display format specified by the user.

1. Complex Criteria

Arbitrarily complex, composite criteria can be constructed from the atomic criteria discussed above. This is accomplished through the use of and, or, and not to link the criteria. The following sample command illustrates this capability:

QUERY 3 C: SHOW C: ALL (PLATFORMS) OK/C:

BETWEEN T/¯A_7: 20 (AND) /¯A_7: 200 C:

NAUTICAL (MILES OF) C: ME OK: /¯10_7

2. Use Of Ellipsis

Often the user can formulate his criteria only one step at a time, using feedback from previous steps as an aid in stating further criteria. Previous versions of Query required the user to employ a cumbersome, time-consuming and redundant step-by-step method of utilizing feedback to assist in formulating criteria. To eliminate such inefficiencies, Query 3's find and show commands are designed so that the system remembers the most recent located set of platforms and allows that set to be used as the initial search domain the next time a command is invoked. Thus it is possible to answer questions such as the following:

Initial Query:

QUERY 3 C:SHOW C: OPCON T/¯A_7: TF33

(DISPLAY FORMAT?) C: MEDIUM OK:

(Query 3 searches the data base and displays those units in TF33).

USE OF ELLIPSES QUESTION NUMBER 1 --

QUERY 3 C: OF (THESE) C: FIND (ALL) C:

CRAFT (WITH) C: OPCON C: EQUAL (TO) T/¯A_7:

TG 33.7 OK/C: (DISPLAY FORMAT?) C:

MEDIUM OK

(Query 3 searches craft located by initial question and displays craft in TG 33.7).

USE OF ELLIPSES QUESTION NUMBER 2 --

QUERY 3 C: OF (THESE) C: FIND (ALL) C:

CRAFT (WITH) C: PERCENT (FUEL LEFT) C:

LESS (THAN) T/¯A_7: 95 OK/C:

(DISPLAY FORMAT?) C: MEDIUM OK:

(Query 3 searches craft found as a result of ellipses question number 1 and provides answer). /¯10_7

The reader should note that in the above example, Query 3 is able to answer the questions utilizing ellipses very rapidly since only those craft satisfying the first condition need be examined.

V. SUMMARY

Today's military environment demands that attention be given to the field of command and control technology. A particular aspect of this field is data base management and retrieval. Query 3 represents an experimental effort using state-of-the-art technology to develop a viable data base inquiry system. Query 3 is a relatively low cost system that required minimal development time since it is based on existing technology. However, these factors should not obscure the possible benefits of developing a Query 3-like system.

Query 3, through it's NLS base, is an excellent demonstration of human interface capabilities with a structured data base. The data base is realistically constructed to represent an actual Navy combat data base. While Query 3 was not developed as an efficient data base management system, it does provide a solid example of a method of approaching a C2 technological problem.

It should also be noted that the placement of Query 3 within the ARPANET enhances the opportunity to utilize Query 3 in a modern communications environment. Furthermore, the ARPANET allows the possibility of Query 3 experimentation in a distributed data base scenario. This is an important aspect of command and control technology since plans for packet switching communications and distributed data bases are soon to become a reality.

In summary, future concepts of world-wide command and control, over-the-horizon targeting, and evolving weapons capabilities demand that reliable, human engineered data base management systems be developed. Query 3 is an excellent start in this direction but more work remains to be done in the laboratory and in the field.

APPENDIX A

QUERY 3 TUTORIAL (BASIC)

I. PURPOSE OF QUERY 3 TUTORIAL

The purpose of this tutorial is to provide a user's guide to learning and using the Query 3 system. This tutorial is designed to present the reader with a logical, simple approach to basic mastery of Query 3 and assumes no prior knowledge of computer techniques. This tutorial is intended to "stand alone"; that is, no other references are necessary to assist in learning the material presented. However, the user should be aware of references that assist in understanding the ARPANET since Query 3 exists on the ARPANET. These references include:

Defense Communications Agency Report NIC 45601, ARPANET Resource Handbook, October 1978.

Holg, C., Joy Of TENEX: 3 ...The Basics, USC/Information Sciences Institute, 1976.

Holg, C., Joy Of TENEX: 4 ...Some Of The Refinements, USC/Information Sciences Institute, 1977.

It should be noted that this tutorial represents solely the opinions of the author. The material contained herein does not necessarily reflect the opinions or policies of Stanford Research Institute, the Naval Ocean Systems Center, or any agencies of the Department of Defense.

II. ABREVIATIONS

Several abreviations are used throughout this tutorial to represent various terminal typewriter keys and other operations. The following list of abreviations is provided for quick reference:

Key/Operation	Abreviation
Carriage Return	C/R
Space/Blank	△
Control Key	CTRL
Escape Key	ESC
Logging Into A System	LOG
Logging Out Of A System	LOGOUT
Prompt Sign	@
Query 3 Herald	QUERY 3 C:

III. COMMANDS

Query 3 provides information to the user by accessing its data base in response to inquiries initiated by the user. These inquiries are composed of commands. Query 3 commands are designed to approximate normal use of the English language and, therefore, are considered relatively easy to utilize. Commands contain one or more options which permit the user to construct inquiries that resemble normal English phrases. Additionally, the system assists the user in constructing inquiries through the use of structure, prompting, and provision of appropriate options upon request. These capabilities will be explored in future sections of this tutorial.

The following list of basic commands is provided for easy reference:

ALTER -- Allows the user to eliminate prompting (not recommended for the novice user).

ASSIGN and DEASSIGN -- Place a commander at or on a specific platform. In addition, this command provides a synonym

capability so that a specific platform (such as the carrier John F. Kennedy) can also be referred to by a synonym (such as JFK).

ATTACH and DETACH -- Change the operational control of a craft.

ESTABLISH and DISESTABLISH -- Group or separate craft to form or abolish a task force.

INCLUDE and REMOVE -- Add craft to or remove craft from an existing task force.

HOW (FAR IS...) -- Determine the distance and bearing of one platform with respect to another or the time that would be required for them to rendezvous.

MOVE -- Move all craft in the data base for a specified number of hours with the appropriate percentages of their fuel being consumed.

PUT -- Place a craft at a port without requiring the user to look up the position of the port and change the craft's position explicitly.

REFUEL -- Refuel an individual craft (by setting its % fuel field to 100).

SHOW -- Is the most versatile command. It allows the user to interrogate the data base on various fields. This makes it possible for the user to ask the data base to display all Russian ships within 500 miles of a specific platform, for example.

UPDATE, DISCARD, and USE -- Update the data base with the changes made since the user started or last updated,

discard those changes, or switch to an alternate data base (see LABEL).

WHERE and MY -- Report or modify the user's position (specified when he entered QUERY 3). This position is used as the reference for data base commands such as "show the doctor nearest me".

OF (These) -- Is the elliptical reference command, which allows the subset of the data base defined by a preceding SHOW or FIND command to be used as the data base for the next command, thereby minimizing processing and user keystrokes.

LABEL and DROP -- Allow the user to specify a label (or delete it) for a subset of the data base for future reference. Later the USE command can permit the user to operate on this subset with QUERY 3 commands.

TEST (Rendezvous Feasibility) -- Test the feasibility of the rendezvous of particular ships at a specified position and time.

INVERT and VERIFY -- Allow the data base to be verified for data consistancy, e.g., the speed of a craft exceeding the maximum speed of its Naval class, and to be inverted.

PRINT -- Display this description of QUERY 3 to the user.

OUTPUT -- Send the contents of the data base to a file or line printer.

CORRECTION and MODIFICATION -- These commands provide a structured modification of the data base in order to maintain data base information integrity. They are

the obvious:

ADD

CHANGE

DELETE [12]

This tutorial provides the user with a series of examples that demonstrate the use of the majority of the basic commands. A more advanced tutorial, designed for persons who have completed basic familiarization with Query 3, is available in Appendix B of this paper.

IV. CONNECTING THE TERMINAL TO AN ARPANET HOST COMPUTER

Authorized users of ARPANET facilities are capable of remotely accessing host computers from a variety of terminals. These terminals may be of the cathode ray tube type or they may be of the teletype-like, hard copy variety. The terminals may be "hardwired" to a TIP (terminal Interface Message Processor) or they may be capable of linking to the TIP via telephone. However, the procedures for accessing a host computer are standard once a terminal is linked to a TIP. The example that follows is based on the use of a computer devices miniterm terminal since the author had easy access to that device during the writing of this tutorial. (Instructions To The User Are Written In Upper Case Letters).

The user should ascertain that the miniterm is ready for use by checking the following features:

Power cord is plugged into 110 volt outlet.

Positions on terminal are set as indicated:

POWER: ON

SPEED: 30

MODE: FULL (DUPLEX)

MODE: STD (TERMINAL MODE)

PARITY: OFF (ERROR RESET)

PARITY: EVEN

Dial the number for the TIP (Terminal Interface Message Processor).

LISTEN FOR A HIGH PITCHED SOUND.

(A busy signal or ringing for more than six times is an unsuccessful attempt at making a connection. Hang up and try again).

PLACE THE TELEPHONE IN THE BACK OF THE TERMINAL.

If the TIP identifier (i.e., NPS TIP 420#5) does not immediately appear:

Type,

e C/R (C/R INDICATES CARRIAGE RETURN)

This operation will alert the TIP to the user's presence. The connection will be completed and the TIP identifier will appear.

At the time of this writing the author had access to a Query 3 demonstration on ARPANET Host Computer 116. If the user has access to Query 3 on another host, that address should be used in the following step.

Type,

@ 0△116 C/R

The "@" symbol alerts the ARPANET that a command follows. The "O" is the command to open a connection. The symbol "△" is used by the author to indicate a blank space. The number "116" is the address of the ARPANET host containing Query 3.

The computer response is "TRYING..." followed by "OPEN" on the next line. This indicates that the connection to the computer is made. The computer will then type its identifier and system information as follows:

ISI-SYSTEM-E, TOPS-20 MONITOR 3A(3116)-1

SYSTEM SHUTDOWN SCHEDULED FOR 28-JAN-80 22:00:00

UP AGAIN AT 29-JAN-80 05:00:00

When this is completed the computer will print the "@" symbol indicating readiness to accept a command.

If the TIP is unable to make the desired connection, a message will be printed indicating the reason. The user should then disconnect the phone and attempt to make a connection at a later time.

Example:

User Types,

e C/R

Terminal Prints,

NPS TIP 420 #5

User Types,

@ O△116

Terminal Prints,

Trying ...

Open

ISI-SYSTEM-E, TOPS 20 MONITOR 3A (3116)-1

SYSTEM SHUTDOWN SCHEDULED FOR 28-JAN-80 22:00:00

UP AGAIN AT 29-JAN-80 05:00:00

V. LOGIN PROCEDURES FOR QUERY 3 DIRECTORY

The Query 3 demonstration utilized in this research was located in a directory called Query 3 on ARPANET Host 116. In order to access that directory the following steps should be followed:

Immediately following the @ symbol

Type,

LOG△QUERY 3 CTRL ESC

The word "LOG" tells the system that the user is entering a directory. "QUERY 3" indicates the directory name. Typing the "CTRL" and "ESC" keys causes the computer to print "(PASSWORD)". This indicates that the computer is ready to accept the password that permits entry into the Query 3 directory.

Type,

(USER WOULD THEN ENTER PASSWORD)

Note that the system does not echo back (that is, print) the password on the terminal. This is a planned feature to protect password security.

Type,

CTRL ESC CTRL ESC C/R

The system responds to the first "CTRL ESC" by printing (ACCOUNT) ACCAT, and to the second "CTRL ESC" by printing

(SESSION-REMARK). Typing the carriage return key indicates completion of the LOGIN procedure. If the LOGIN procedures are correctly completed, the computer prints a job number and other information on the next line. The following sample indicates the format:

JOB 55 ON TTY 162/ 28-JAN-80 09:13:46

The computer then automatically moves to the next line and prints the "@" symbol indicating readiness to accept commands.

Example:

User types,

LOG△QUERY 3 CTRLESC (PASSWORD) CTRLESC CTRLESC C/R

Terminal prints,

(Password)(Account) ACCAT (Session-Remark)

JOB 55 on TTY 164 28-JAN-80 09:13:46.

@

VI. ENTERING THE QUERY 3 SYSTEM

The Query 3 system design is based on the properties of a computer software system called Online System (NLS). To enter the Query 3 system, the user must first enter NLS which is programmed (in this case) to activate Query 3. The procedures involved are quite simple and are illustrated in the following example:

User types,

NLS C/R

Terminal prints,

IDENT =

User types,

Q C/R

Terminal prints,

BASE C: SIMULATE (Terminal Type) C: T_1 (Terminal) OK:

BASE C: GOTO (Subsystem) C: PROGRAMS OK:

PROG C: DELETE C: ALL (Programs In Buffer) OK:

PROG C: LOAD C: PROG T/$\overline{_}$A_7: LOAD 3

LOADING USER PROGRAM

Don't execute via run program command use

GOTO Subsystem Command

Loading User Program

Subsystem Load 3 Now Available (attached)

PROG C: QUIT OK/C:

BASE C: GOTO (Subsystem) C: LOAD 3 OK:

LOAD C: LOAD C: NOVICE (QUERY 3) OK:

LOAD C: GOTO (Subsystem) C: QUERY 3 OK:

THE START OF XQINIT

XQINIT IN PARSE MODE

END OF XQINIT

START OF XQVERSION

THIS IS VERSION 3.6 OF QUERY DATED 29-APR-77.

THE END OF XQVERSION

(Do you want a description of Query?) C:

At this point the user has the option of viewing a brief description of Query 3. The user should type "Y" for yes or

"N" for no. The system will complete typing the words yes or no and will print "OK:". This symbol is used as a safeguard against user error, throughout the system. If the user is satisfied that his command (yes or no) is correct, he simply strikes the carriage return and the order is executed. There are several methods for recovery from keyboard errors which will be discussed later in this tutorial.

Example (continued):

User types,

N

Terminal prints,

0 OK:

User types,

C/R

Terminal prints,

(What is your position?) T/_A_7:

Here the user is required to input a geographical position given in latitude and longitude. This input determines whether the system will use the Atlantic or Pacific data base and establishes the user's reference position. The symbol "T/_A_7:" is used by Query 3 to indicate that user generated text (such as latitude and longitude) is required.

Latitude and longitude entries must consist of 3 to 5 digits and must be followed by "N" or "S" (latitude) and "E" or "W" (longitude).

Example (continued):

User types,

34-00N/124-00W

Terminal prints,

Loading Pacific Data Base

You are at 34-00N 124-00W in the Pacific

Query 3 C:

The herald "Query 3" indicates that the system is now ready to accept inquiries about the data base.

Note: Query 3 is capable of using various data bases which may be accessed by the "USE" command. This is discussed in Section XV.

VII. QUERY 3 USER ASSISTANCE AND KEYBOARD ERROR RECOVERY

Query 3 is designed to offer the user immediate assistance when the user is in doubt as to what alternatives are available. This capability is based on features inherent in the Online System. The user may obtain a listing of alternative commands at any time by typing a question mark (?).

For example, assuming that the user has just entered the Query 3 system and is unsure of the command alternatives available, he would proceed as follows:

Terminal has printed,

QUERY 3 C:

User types,

?

Terminal prints,

Current Alternatives are:

Add	GOTO (subsystem)	Refuel
<>Alter	Help	<>Remove (from task...)
<>Assign	How (far is platform)	Show
<>Attach (opcon)	Include (in task...)	<>Syntax (of command)
Change	Jump (to)	Test (feasibility...)
<>Chop	Label	Use (as data base)
<>Compute	<>List	Version (Of Query)
<>Deassign (command)	<>Move (all craft f...)	Where (am I?)
Delete	My (position is)	;
<>Detach (opcon)	Of (these)	<
Disestablish (tas...)	<>Output	>
Establish	<>Print	CTRL-Q : HELP
Execute (command in)	<>Put (craft)	CTRL-S : SYNTAX
Find	Quit	REPEAT

This list is a compilation of Query 3 commands available to the user at this level. The symbol <> indicates that the first character of those commands so marked is a space. The reader should note that the CTRL-Q and CTRL-S commands are not currently available and should not be used.

Query 3 is designed to recognize commands after receiving only the first character or the first several characters of the command word. This allows the user to type only a minimum number of characters and places the majority of the work load on the system. For example, let us assume that the user has reviewed the list of alternatives and has decided to use the "SHOW" command:

User types,

S

Terminal prints,

How C:

The system is now ready for the user's next input. If uncertainty exists concerning the next input, the user may again type a question mark.

User types,

?

Terminal prints,

Current Alternatives are:

```
<>Active (surveilla...)   Group (labelled)     <>The (one)
  All                     Opcon                <>Track
<>Attack (threats t...) <>Passive (surveill...)<>Unknown
  Class                   Platform               Us
<>Craft                 <>Ports                  CTRL-Q : HELP
  Flag                    Ship                   CTRL-S : SYNTAX
<>Foreign                 Task
```

At this point the user may type in the appropriate alternative and continue building his command. The use of the question mark to obtain information concerning alternatives may be continued if required.

Keyboard error recovery in Query 3 is a relatively simple matter. The following commands apply:

CTRL-A Deletes the last character.

CTRL-W Deletes the last word.

CTRL-X Deletes the command presently being typed in.

CTRL-O Stops information printout.

Example:

User has entered the following:

QUERY 3 C: SHOW C: PLATFORM C:

and wishes to delete the entire command.

User types,

CTRL-X

Terminal moves to next line and types,

QUERY C:

The system has cancelled the previous command and is now ready to receive a new input.

Note: on the computer devices miniterm the "RUB" key performs the same function as "CTRL-X".

VIII. MY (POSITION IS)/WHERE (AM I?) COMMANDS

The My (position is) command is utilized to alter the user's reference position within the system. Where (am I?) is used to obtain a report of the user's position within the system.

Example /Where (am I?)/:

User types,

W

Terminal prints:

Here (am I?) OK:

User types,

C/R

Terminal prints,

You are at 34-00N 124-00W in the Pacific;

The Where (am I?) command is useful to the user as a means of obtaining confirmation of his reference position. Confirmation may be necessary since the user's reference position can be utilized as a portion of certain inquiries.

Example /¯My (position is)_7:

User types,

M

Terminal prints,

Y (position is) T//¯A_7:

User types,

34-00N/78-00W

Terminal prints,

Loading Atlantic data base.

You are at 34-00N 78-00W in the Atlantic;

The My (position is?) command provides the user with the flexibility to move to different reference positions within a particular ocean data base or to move between ocean data bases. This feature is important to the user who needs to establish reference positions close to various platforms within the data base or who needs to move between the Atlantic and Pacific data bases in order to maintain a two ocean picture.

Note: If the user wishes to exit QUERY 3 at this time, refer to Section XVI for instructions.

Now that the user has become familiar with the methods of human interaction with Query 3, presentation of examples will be accomplished in a more concise format. The following conventions will be used in future sections of this tutorial:

-- User input will appear in underlined, capital letters.

Example = F

-- System output will appear in lower case letters enclosed by quotes.

Example = "ind (all)c:"

IX. THE SHOW COMMAND

The Show command is one of the most versatile commands contained within Query 3. This command allows the user to interrogate the data base concerning various fields of information and is particularly valuable when the user is formulating criteria. The Show command's alternatives (obtained by entering a "?" character after the "Show") are presented in the following list:

Active (surveillance)
All
Attack
Class
Craft
Flag
Foreign
Group (labelled)
Opcon
Passive (surveillance)
Platform
Ports
Ship
Task
The (one)
Track
Unknown
US

Show command alternatives provide the user with a great deal of flexibility. The Show command enables the user to establish data base contents concerning individual ships or

ports, active and passive surveillance threats, operational control data, groups of ships or ports, etc.

Examples:

"query 3 c:" S"how c:"

(At this point the user is required to choose one of the "show" command alternatives.)

P"latform t//a7:"

(The user must type in the name of a specific platform or port.)

TURNER C/R

"(display format?)c:

(The user must indicate the type of display format desired. Formats available are:

Long - includes all data available concerning platform.

Medium - includes only a portion of data.

Short - lists only platform name, flag and position.

Tabular - lists platform names only.

None - returns user to Query 3 herald.

Graphical - not to be used.)

L"ong ok:" C/R

```
"name=richmond k.turner synonyms=turner plat=s flag=us
 class=leaky cat=nav type=clg hul=20 conam=cdr d. flowers
 opcon=tg21.4 doctr=y pcfuel=30.00 ptp=50-31n 8-03w
 ptc-004 pts=21.0 lgt=533 beam=55 draft=25 displ=7800
 endur=45 ftp=lj mcs=34.0 mcm=1800 ncs=16.1 ncm=5000
 guns=4 gunsize=3"/50 tsize=13 tubes=6 tnom=mk32 aslch=
 asroc lnchrs=1 missl=terrier mislnch=4 misrng=25
 query 3 c:"
```

This example has demonstrated a simple use of the show command. The next examples will demonstrate the command's ability to retrieve information about groups of platforms.

"query 3 c:" S"how c:" C"lass t//a7:"

(The user must type in the name of the class desired.)

ADAMS C/R

"ok/c:" C/R

"(display format?) c:" S"hort ok:" C/R

"benjamin stoddart us 18-24n 70-26w

berkeley us 40-51s 39-48e

buchanan us 40-00s 39-24e

cochrane us 24-42n 34-09w

conygham us 30-07n 49-39e

goldsborough us 50-00s 40-26e"

"richard e. byrd us 45-03n 13-21e

sampson us 24-03n 35-02w

semmes us 29-55n 48-56e

tattnal us 34-26n 39-05e

waddell us 50-23n 32-30w

query 3 c:"

Note the difference between the long and short display formats. The next example illustrates the tabular format.

"query 3 c:" S"how c:" A"ll(platforms) ok/c:"

W"ithin t/¯a7:"

(The user must enter a number which will represent either nautical miles or hours.)

500 C/R "c:"

(Type "N" for nautical miles or "H" for hours.)

N"autical (miles of) c:" P"latform t/¯a7:"

(Type the name of a platform.)

WADDELL C/R

"(display format?) c:" Tabular ok:" C/R

```
"detailed search of 307 records is 7% complete; 0 records
 found so far.
 detailed search of 307 records is 16% complete; 0 records
 found so far.
 detailed search of 307 records is 27% complete; 0 records
 found so far.
 detailed search of 307 records is 39% complete; 0 records
 found so far.
 detailed search of 307 records is 52% complete; 2 records
 found so far.
 detailed search of 307 records is 65% complete; 4 records
 found so far.
 detailed search of 307 records is 78% complete; 4 records
 found so far.
 detailed search of 307 records is 88% complete; 5 records
 found so far.
 josephus daniels  harry e. yarnell  truett   william h. sta...
 panamerican fl...  unk727     england
 query 3 c:"
```

/Note - the system prints the "detailed search..."statement as a progress indication for the user./

The show command is an extremely useful Query 3 operation. Standing alone it offers the user a wide range of options. When utilized with other Query 3 features (such as, ellipses and complex inquiries -- discussed in sections XIV and XIII), the show command increases in user value.

X. THE FIND (ALL) COMMAND

The Find (all) command is another very versatile command. It's main value to the user is in determining groups of ships or ports with specific, similar characteristics. There are four alternatives initially available in the Find (all) command. They are:

Classes	Platforms
Craft	Ports

The fact that there are only four alternatives initially available may seem to be relatively constrained. However, the follow-on alternatives are numerous and provide a wide range of input parameters. These alternatives are:

Active (surveilla...)	Ftp	Not
Aslnch	Guns	Opcon
At (range of)	Gunsize	Own
Attack (threat to...)	Gwt	Passive (surveill...)
Beam	Hull	Percent (fuel left)
Between	Labels	Plat
Cargo (type)	Lgh	Position
Category	Lnchrs	Quantity (of cargo)
Class	Mcm	Satisfying
Conam	Mcs	Speed
Course	Mislnch	Synonym
Displ	Misrng	Tnom
Doctor	Missl	Tsize
Draft	More (than)	Tubes
Dwt	Name	Type
Endur	Nat	Within
Failing	Ncm	(
Farthest (in)	Ncs	
Flag	Nearest (in)	

The Find command is particularly useful when attempting to locate a similar group of platforms. It is also appropriate when used in conjunction with the ellipsis technique

and when forming complex criteria. /These operations will be presented in later Sections of this tutorial./

Example:

```
"query 3 c:" F"ind (all) c:" P"latforms (with) c:"
△MISS"l c:" E"qual (to) t/a/:" TARTAR C/R
"ok c:" C/R

"(display format?) c:" T"abular ok:" C/R

"detailed search of 307 records is 22% complete; 0 records
 found so far.
 detailed search of 307 records is 36% complete; 5 records
 found so far.
 detailed search of 307 records is 45% complete; 9 records
 found so far.
 detailed search of 307 records is 54% complete; 10 records
 found so far.
 detailed search of 307 records is 54% complete; 10 records
 found so far.
 detailed search of 307 records is 58% complete; 11 records
 found so far.
 detailed search of 307 records is 66% complete; 12 records
 found so far.
 detailed search of 307 records is 72% complete; 16 records
 found so far.
 detailed search of 307 records is 75% complete; 16 records
 found so far.
 detailed search of 307 records is 82% complete; 17 records
 found so far.
 detailed search of 307 records is 90% complete; 18 records
 found so far.
 detailed search of 307 records is 96% complete; 18 records
 found so far.
 audance   benjamin stoddart   berkeley   bouvet
 buchanan   cochrane   conyngham   dupetit thouars
 goldsborough   kersaint   lut jens   molders
 richard e. byrd   rommel   sampson   semmes
 tattnall   waddell
 query 3 c:"
```

The user now has a listing of all ships within the data base that are equipped with TARTAR missiles. The Find command can be used to collect further information concerning the data base by exercising other options available. Further examples of the Find command are available in Sections XIV (Ellipsis) and XIII (Complex commands).

Note: If the user wishes to exit Query 3 at this time, refer to Section XVI.

XI. THE HOW (FAR...) COMMAND

The How (far...) command is used to determine the distance and bearing of one platform with respect to another or the time that would be required for those platforms to rendezvous. The following examples illustrate both aspects of this command.

Examples:

"query 3 c:" H"ow (far is platform) t/¯a_7:" BERKELEY C/R

"(from) c:" P"latform t/¯a_7:" SAMPSON C/R

"(in)c:"

/¯The user msut type "H" for hours or "N" for nautical miles_7

N"autical (miles) ok:" C/R

"bearing of 133 degrees at a range of 5695 nautical miles

query 3 c:"

The preceeding portion of the example demonstrates the distance and bearing feature of the How (far...) command. The remaining portion illustrates the rendezvous timing feature of the command.

"query 3 c:" H"ow (far is platform) t/¯a_7:" ROMMEL C/R

"(from) c:" P"latform t/¯a_7:" COCHRANE C/R

"(in) c:" H"ours ok:" C/R "(with first platform) c:"

/¯The user must indicate if the platform is moving or stationary by typing an "M" or "S"._7

S"tationary ok:" C/R

/If the moving choice was indicated the user would be given speed options./

"(and second platform moving at) c:"

/The alternatives are Current (speed), Economical (cruising speed), Maximum, or Standard (speed)/.

C"urrent (speed) ok:" C/R

"122 hours"

"query 3 c:"

XII. THE TEST (RENDEZVOUS FEASIBILITY) COMMAND

This command is used to test the feasibility of the rendezvous of particular ships at a specified position and time. The value of this command is that the user may check the appropriateness of a movement order prior to issuance of that order. The command is flexible and permits testing of rendezvous feasibility for more than two ships.

Example:

"query 3 c:" T"est (feasibility of rendezvous at position) t//a/:"

56-00N/34-00W C/R

"(craft) t//a/:" STANDLEY C/R

"(finished?) c:"

/The user has the option to type "Y" for yes or "N" for no to indicate if another platform is to be considered in the rendezvous test./

N"o ok:" C/R

"(craft) t//a/:" WAINWRIGHT C/R

"(finished?) c:" Y"es ok:" C/R

"(proceeding at) c:"

/¯The user has the option of indicating a desired rendezvous speed from the following choices: Current (speed), Maximum (cruising...), Economical (cruis...), and Standard (speed f...)./

E"conomical (cruising speed) ok:" C/R

"(time constraint?)"

/¯The user must indicate if a time constraing is required by typing a "Y" for yes or "N" for no./

N"o ok:" C/R

"name=william h. standley range=627 hours=39 pcfuel=25.61
name=wainwright range=874 hours=55 pcfuel=0 with 764nm to go.
Rendezvous is infeasible.
query 3 c:"

In this case the rendezvous is not feasible because the Wainwright will run out of fuel prior to reaching the rendezvous position. The following case depicts a successful test.

"query 3 c:" T"est (feasibility of rendezvous at position)
t/¯a/:" 56-00N/34-00W C/R
"(craft) t/¯a/:" STANDLEY C/R
"(finished?) c:" N"o ok:" C/R
"(craft) t/¯a/:" WADDELL C/R
"(finished?) c:" Y"es ok:" C/R
"(proceeding at) c:" E"conomical (cruising speed) ok:" C/R "(time constraint?) c:" Y"es ok:" C/R
"(rendezvous within how many hours?) t/¯a/:" 45 C/R

"name=william h. standley range=627 hours=39 pcfuel=25.61"
"name=waddell range=341 hours=21 pcfuel=95.15
craft can rendezvous in 39 hours.
(initiate the rendezvous?) c:"

/¯The user must enter "Y" for yes or "N" for no./

N"o ok:" C/R

"query 3 c:"

Note: If the user wishes to exit Query 3 at this time, refer to Section XVI for instructions.

XIII. COMPLEX COMMANDS

Arbitrarily complex, composite criteria can be constructed from the atomic criteria contained within the data base. This feature is particularly useful when constructing inquiries that require answers concerning the platforms between an inner and outer range limit or when utilizing multiple criteria with the Find command. The criteria are linked through the use of "and", "or", and "but" to link the criteria. The following examples illustrate this capability:

Examples:

"query 3 c:" F"ind (all) c:" P"latforms (with) c:"

T"ype c:" E"qual (to) t/¯a/:" CLGN C/R

"ok/c:" O"r c:" T"ype c:" E"qual (to) t/¯a/:"

SSN C/R "ok/c: C/R

"(display format?) c:" S"hort ok:" C/R

"texas us clgn 44-12n 14-15e
virginia us clgn 59-17n 28-01w
drum us ssn 29-53n 31-04w
finback us ssn 49-23n 71-40w
groton us ssn 36-21n 21-03e
hammerhead us ssn 34-53n 9-47e
l.indomptable fr ssn 22-49n 12-22w
l.mendel rivers us ssn 80-39n 70-52w
memphis us ssn 0-17s 45-30e
omaha us ssn 14-53s 12-16e
ray us ssn 0-02s 70-19w
sandlance us ssn 50-12n 70-29w
tunny us ssn 28-55n 60-28w
query 3 c:"

/¯Warning - The construction of complex commands is limited to using either "and" or "or" in a single list. The use of "and" and "or" together will abort the query./

"query 3 c:" F"ind (all) c:" P"latforms (with) c:" T"ype c:" E"qual (to) t//¯a/:" CLG C/R
"ok/c:" B"ut c:" F"lag c:" E"qual (to) t//¯a/:" US C/R "(display format?) c:" T"abular ok:" C/R

"biddle	dale	england	fox
gridley	halsey	harry e. yarnell	horne
josephus daniels	jouett	leahy	reeves
richmond k. turner	sterett	wainwright	william h. sta..
worden			

query 3 c:"

XIV. OF (THESE) COMMANDS

The "of these" command is an elliptical reference command which allows the subset of the data base determined by a

preceeding "show" or "find" command to be used as the data base for the next command. This presents the user with the advantage of being able to quickly and efficiently narrow a large field of data to a smaller range of desired information. This is particularly valuable when the user is initially unsure of his data requirements and feedback is necessary to reduce ambiguity. The following examples illustrate the use of ellipsis:

Example:

Assume that the user has initiated the query "show c: opcon t/¯a7: tf 67" and obtained the following tabular response on the terminal,

"dwight d. eise...	nimitz	ainsworth	fox
reeves	valdez	dale	biddle
miller	pharris	richard e. byrd	texas
conygham	semmes	goldsborough	

query 3 c:"

"query 3 c:" O"f (these) c:" F"ind (all) c:"

C"raft (with) c:" ΔPE"rcent (fuel left) c:"

L"ess (than) t/¯a7: 95 C/R "ok/c:" B"ut c:"

ΔPE"rcent (fuel left) c:" G"reater (than) t/¯a7:"

50 C/R "ok/c:" C/R

"(display format?) c:" T"abular ok:" C/R

"fox richard e. byrd"

"query 3 c:"

Now the user has determined which ships in Task Force 67 are not within desired fuel limits, assume that he needs to

know which refueling capable ships are in the data base and which one of these ships is closest to the Byrd. The Query "show c: all (platforms) ok/c: type t /¯a_7: refuel ok.c" produces the following response:

"alatyr ur ao 4-10s 48-24e
andrey ur ao 3-48s 48-10e
caloosahatchee us ao 24-00n 34-50w
desna ur ao 4-18s 47-44e
dwight d. eisenhower us cvan 44-56n 15-18e
forrestal us cv 17-56n 70-10w
john f. kennedy us cva 60-06n 29-51w
kiev ur cv 65-52n 0-40e
kittyhawk us cva 25-03n 34-09w
leningrad ur cv 32-36n 30-07e
minsk ur cv 73-05n 2-49w
moskva ur cv 32-58n 30-05e
nimitz us dvan 38-15n 30-05e
ponchatoula us ao 17-22n 70-12w
ranger us cv 17-32n 70-29w
truckee us ao 49-28n 70-41w
volkhov yr ai 32-44n 30-14e
query 3 c:"

"query 3 c:" O"f (these) c:" S"how c:" △TH "e(one) ok/c:"
N"earest (in) c:" N"autical (miles (to) c:"
P"latform t/¯a_7:" BYRD C/R
"(display format?) c:" L"ong ok:" C/R

"detailed search of 17 records is 23% complete; 0 records found so far.
detailed search of 17 records is 52% complete; 0 records found so far.
detailed search of 17 records is 64% complete; 0 records found so far.

detailed search of 17 records is 82% complete; 0 records found so far.
name=dwight d. eisenhower synonyms=ike plat=s flag=us class=nimitz cat=nav type=cvan hul=69 conam=capt j. russel opcon=ctf67 doctr=y pcfuel=100.00 ptp=44-56n 15-18e ptc=170 pts=27.0 lgh=unk beam=unk draft=unk displ=unk endur=999 ftp=1n mcs=35.0 mcm=inf ncs=20.0 ncm=inf guns=4 gunsize=5"/38 missl-bpdms mislnch=3 misrng=90
query 3 c:"

XV. THE USE (AS DATA BASE) COMMAND

This command is designed to permit the use of NLS files as a data base for Query 3. It's properties are advantageous in that it permits utilization of various data base files which may contain specific information, which by their nature, may require separate maintenance.

Example:

"query 3 c:" U"se (as data base) c:" F"ile t/¯a_7:"

/¯The user must enter the name of the desired data base file._7

C.DB.1 C/R

"(what is your position?) t/¯a_7:"

34-00N/78-00W C/R

"loading atlantic data base"

"you are at 34-00n 78-00w in the pacific;"

"data base is file query 3 c.db.1"

" query 3, c.db.1, "

"query 3 c:"

XVI. LEAVING QUERY 3 AND LOGGING OUT OF THE ARPANET

Exiting from the Query 3 system requires use of the quit command. The following example illustrates the method required:

Terminal has printed,

Query 3 c:

User types,

Q

Terminal prints,

UIT OK/C:

User types,

C/R

Terminal prints,

(Do you want to update the data base permanently?) c:

(The user must indicate "no" as the response to this question since the Query 3 demonstration data base is not permitted to be updated by users.)

User types,

N

Terminal prints,

O ok:

User types,

C/R

Terminal prints,

load c:

User types,

Q

Terminal prints,

uit ok/c:

User types,

C/R

Terminal prints,

base c:

User types,

Q

Terminal prints,

uit ok/c:

User types,

C/R

Terminal prints,

@

The "@" symbol indicates that the user has left the Query 3 and NLS subsystems and has been returned to the ARPANET Operating System.

Leaving the ARPANET requires the use of the LOGOUT command.

Terminal has printed,

@

User types,

LOGOUT C/R

Terminal prints,

killed job /#/, user /name/, account /name/, tty /#/,

at /date//time/, used /time/ in /time/

closed

When the word "closed" has been printed the user may disconnect the phone from the terminal and move the terminal power switch to off.

APPENDIX B

QUERY 3 TUTORIAL (ADVANCED)

I. This tutorial is intended to be utilized by individuals who have attained basic familiarity with the Query 3 system. The contents of this tutorial deal with some of the more complicated portions of Query 3 and is written in a manner that assumes prior knowledge of basic system operation. This publication can be used as a follow-on to the basic Query 3 tutorial contained in Appendix A.

Topics to be presented include data base maintenance commands, graphics, and alteration of user interface. The material is presented with an explanation of the commands and user instruction is provided through practical examples. The examples are designed to guide the user through the use of the commands presented. Examples will be presented in a short, concise format utilizing the following conventions:

-- User input will appear in underlined capital letters.

Example = F

-- System output will appear in lower case letters enclosed by quotes.

Example = "ind (all) c:"

-- Carriage return will be indicated by C/R

-- Blank space will be indicated by a triangle (△).

II. QUERY 3 GRAPHICS

The Online system (NLS) foundation upon which Query 3 is constructed contains a comprehensive software package for

generating and viewing line drawing consisting of mixed text and graphics. The user is able to access the graphics capability by means of an option available in both the find and show command. This option appears when the user is determining his choice of display format. By typing the letter "G", the graphical option is selected. The user is then able to select short, medium, or long textual options.

The graphics option is accessible only from terminals equipped with the necessary graphical display devices.

The following examples are presented to illustrate the graphics capability. Remember that user inputs appear in underlined, capital letters and system output appears as lower case letters enclosed in quotes.

Examples:

/¯Note: These examples are based on Pacific Data Base./

"query 3 c:" F"ind (all) c:" C"raft (with) c:"

ΔAC"tive (surveillance threat to) t/¯a/:"

KITTYHAWK C/R

"(display format?) c:" G"raphical ok:" C/R

"c:" S"hort ok:" C/R

The display generated is illustrated in Figure 9. The next example utilizes the information that is present in the preceeding display through use of the elliptical reference command.

"query 3 c:" O"f (these) c:" F"ind (all) c:"

C"raft (with) c:" ΔPA"ssive (surveillance threat to)

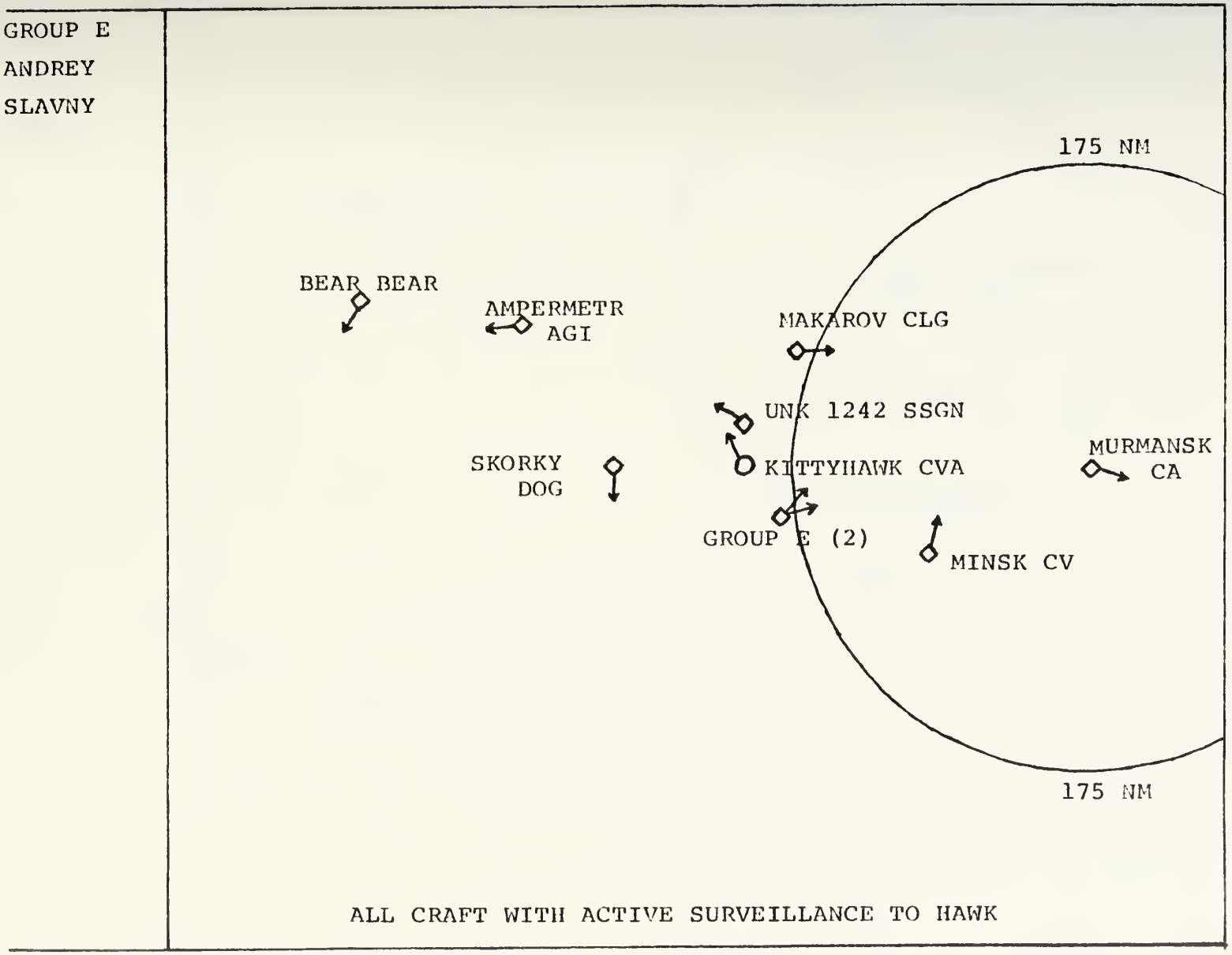


FIGURE 9. Query 3 Graphics 1

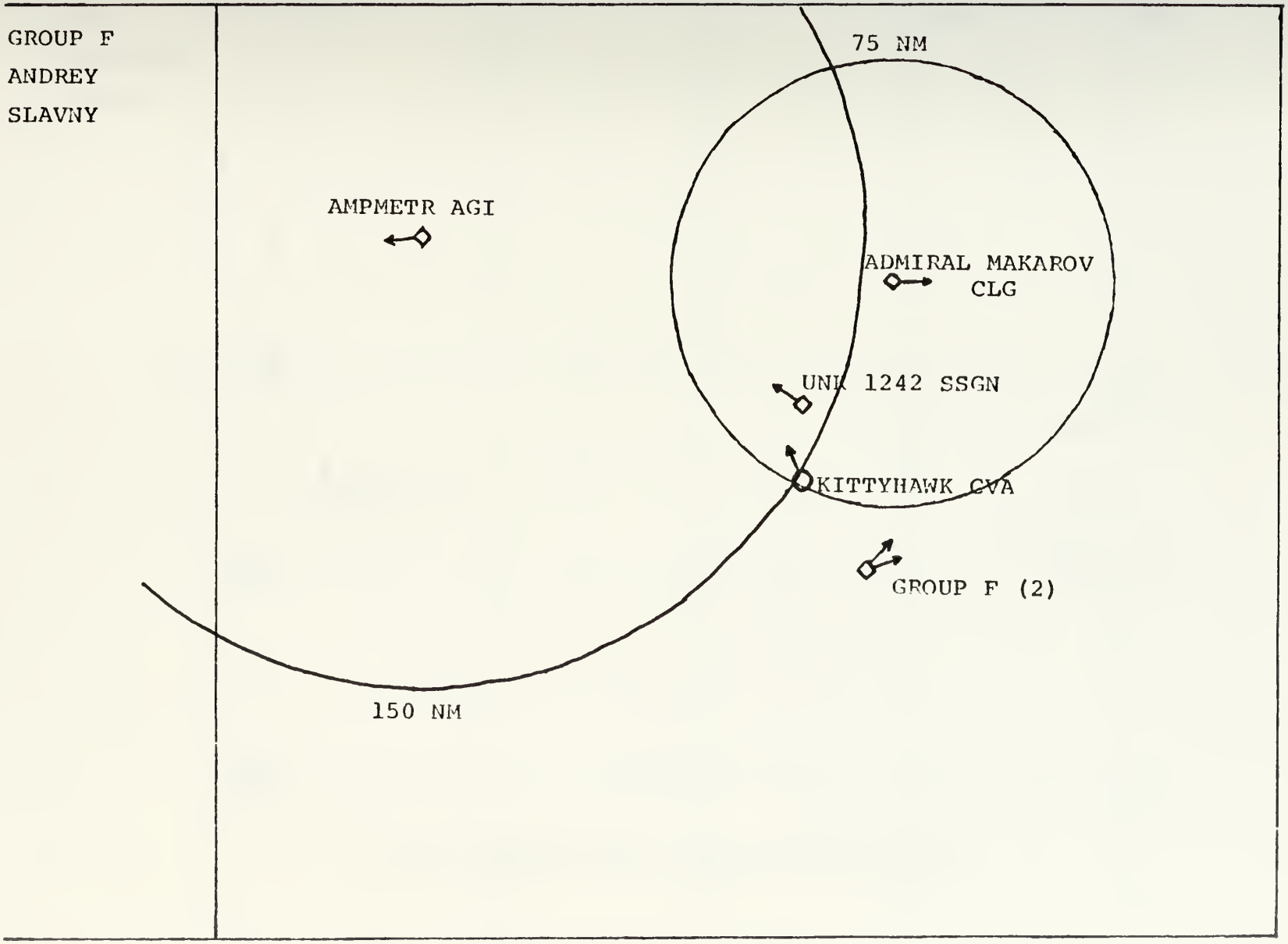


FIGURE 10. Query 3 Graphics 2

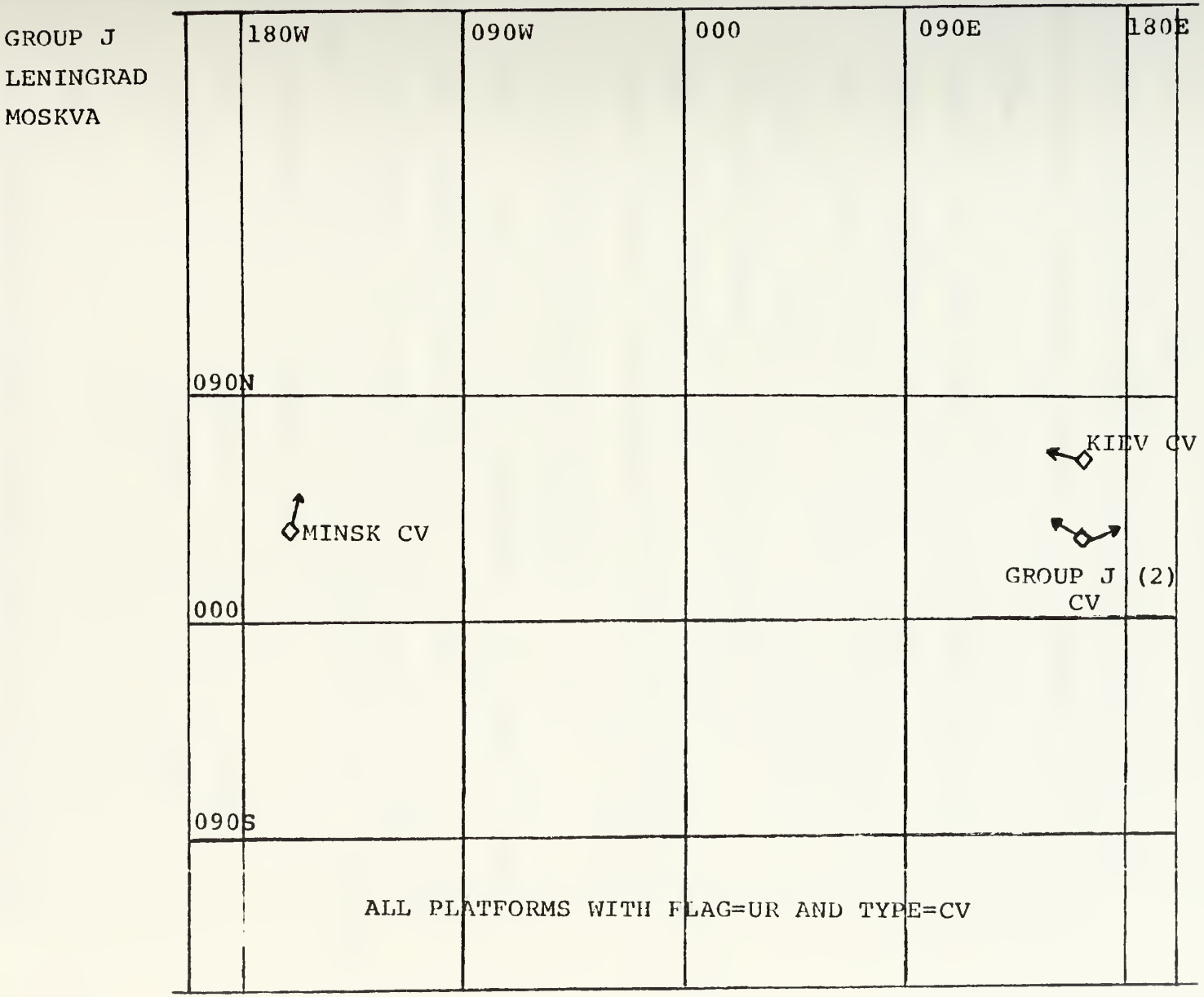


FIGURE 11. Query 3 Graphics 3

t/_a_7:" KITTYHAWK C/R

"(display format?) c:" G"raphical ok:" C/R

"c:" S"hort ok:" C/R

The result of this query is shown in Figure 10. The next example illustrates the use of a complex command with graphics.

"query 3 c:" F"ind (all) c:" P"latforms (with) c:"

F"lag c:" E"qual (to) t/_a_7:" UR C/R

"ok/c:" A"nd c:" T"ype c:" E"qual (to) t/_a_7:"

CV C/R "ok/c:" C/R

"(display format?) c:" G"raphical ok:"

C/R "c:" S"hort ok:" C/R

Figure 11 shows the graphical results of this Query.

III. DATA BASE MAINTENANCE COMMANDS

Query 3 contains various commands that permit a user to to change existing data fields or to create new data. This is an advantage in an operational sense since the user is capable of updating data on an "as required" basis. Commands to be presented in this section are:

Add	Assign	Change
Delete	Include	Label
List	Move	Put
Refuel		

The following examples are presented to illustrate the use of these commands. The examples are based on the assumption that the user is in the Pacific Data Base.

Examples:

LABEL COMMAND - This command is used to specify a label for a subset of the data base. To illustrate the use of this command, assume that the user has displayed CTG 77.2 consisting of Biddle, Dale, Connole, England, Henery B. Wilson, and Hassayampa and desires to work with this group as a subset of the data base with the label "Numero Uno".

"query 3 c:" ΔL"abel c:" T"hese t/¯a¯7:" NUMERO UNO C/R

"query 3 c:"

Confirmation of the label may be obtained by using the show command as follows:

"query 3 c:" S"how c:" G"roup (labelled) t/¯a¯7:"

NUMERO UNO C/R

"(display format?) c:" T"abular ok:" C/R

"biddle dale connole england

henry b. wilson hassayampa

query 3 c:"

ADD COMMAND - This command is used to add new information to the data base. The following options are available in this command:

Cargo (to merchant...)	Merchant
Craft	Naval
Dimensions (to class)	Port
Weaponry (to Naval...)	

For the purposes of illustration, assume that it is desired to add a new craft to the data base.

"query 3 c:" A"dd c:" C"raft"

"(fields may be specified as unknown -- unk)"

"(name=) t/¯a¯7:" CRONKITE C/R

"(platform=) c:" S"urface (craft) ok:" C/R

"(flag -- e.g. us=) t/¯a¯7:" UR C/R

"(class=) t/¯a¯7:" RUSKI C/R

"(category=) c:" N"aval ok:" C/R

"(type -- e.g. tnkr=) t/¯a¯7:" CG C/R

"(hull number=) t/¯a¯7:" 12 C/R

"(commanding officer=) t/¯a¯7:" CDR. BATTENOV C/R

"(doctor?) c: Y"es ok:" C/R

"(percent fuel left=) t/¯a¯7:" 100 C/R

"(position=) t/¯a¯7:" UNK C/R

"(bearing=) t/¯a¯7:" UNK C/R

"(present speed=) t/¯a¯7:" 25 C/R

"query 3 c:"

To ascertain that the new platform has in fact been entered, the user may utilize the show command to display the new data.

"query 3 c:" S"how c:" P"latform t/¯a¯7:" CRONKITE C/R

"(display format?) c:" L"ong ok:" C/R

"name=cronkite plat=s flag=ur class=ruski cat=nav type=cg
hul=12 conam=cdr b. battenov doctr=y pcfuel=100.00
ptp=unk ptc=unk pts=25.0 lgh=unk beam=unk draft=unk
displ=unk endur=unk ftp=unk mcs=unk mcm=unk ncs=unk
ncm=unk
query 3 c:"

CHANGE COMMAND - The change command is used to update data fields for Craft, Merchant, Naval, Port, and Track History records. The data fields which may be altered are:

Cargo Type	Flag	Quantity Of Cargo
Category	Hull Number	Speed
Class	Name	Type
Conam	Percent Fuel Left	
Course	Plat	
Doctor	Position	

The following examples shows the change command applied to the craft Cronkite.

"query 3 c:" C"hange c:" C"raft t/ a 7:" CRONKITE C/R

"(specify parameters to be changed)"

"ok/c:" ΔPE"rcent (fuel left) t/ a 7:" 57 C/R

"ok/c:" ΔCOU"rse t/ a 7:" 340 C/R

"ok/c:" C/R

"query 3 c:"

Use of the show command to display the craft Cronkite would demonstrate that the fields PCFuel and PTC were changed to 57 and 340 respectively.

INCLUDE COMMAND - This command is used to add a ship to a particular opcon. Since the newly created Cronkite has not been assigned an Opcon it will be used to illustrate the use of the "include" command.

"query 3 c:" I"nclude (in task force) t/ a 7:" TI C/R

"(craft) t/ a 7:" CRONKITE C/R

"(finished?) c:" Y"es ok:" C/R

"(issue orders?) c:" N"o ok:" C/R

"query 3 c:"

The user may insure that Cronkite's Opcon has been changed by displaying the craft in the long format and noting the Opcon data field.

ASSIGN COMMAND - This command is used to assign synonyms to platforms which may be used in place of the platform's actual name. This feature is useful when a platform with a rather lengthy name is to be accessed frequently. For instance, the John F. Kennedy may be referred to as JFK, thereby reducing the user's typing requirements.

In the next example, the user will assign a synonym to the platform Cronkite.

"query 3 c:" ΔAS"sign c:" S"ynonym t/¯a_7:"

WALTER C/R

"(to platform) t/¯a_7:" CRONKITE C/R

"query 3 c:"

The user may ensure that the synonym has been entered by using the show command.

"query 3 c:" S"how c:" P"latform t/¯a_7:" WALTER C/R

"(display format?) c:" T"abular ok:" C/R

"cronkite"

"query 3 c:"

MOVE COMMAND - This command is used to move the ships in the data base with appropriate percentages of fuel consumed.

It is activated as follows:

"query 3 c:" ΔM"ove (all craft for how many hours?) t/⌐a¬:"
22 C/R

"(refuel automatically?) c:" N"o ok:" C/R

"query 3 c:"

The user may check the results of this operation by displaying a craft and comparing its current position and fuel percentage with its past data in those fields.

REFUEL COMMAND - This command is used to change the PCFuel data field for one or more ships. The following example brings the fuel percentage of the Cronkite, which had previously been changed to 57, back up to 100 percent.

"query 3 c:" R"efuel (craft) t/⌐a¬:" CRONKITE C/R

"(finished?) c:" Y"es ok:" C/R

"query 3 c:"

PUT COMMAND - This command is used to place a craft in a specific port without requiring the user to look up the geographical position of that port or changing the craft's explicit position data field.

"query 3 c:" P"ut (craft) t/⌐a¬:" CRONKITE C/R

"(at port) t/⌐a¬:" NORFOLK C/R

"(speed and course change?) c:" N"o ok:"

"query 3 c:"

Norfolk's geographical position is 37-00N 76-00W. The user may verify that this is now Cronkite's position by the use of the "show" command.

DELETE COMMAND - This command is used to delete portions of the data base. The command is flexible and permits the following alternatives:

Active (surveilla...)	Group (label)
Attack (threat de...)	Passive (surveill...)
Changes (to data...)	Platforms
Classes	Ports
Craft	Search (criterion)
Diagram (labelled)	Synonym

The following example uses the "changes (to data)" alternative and deletes all of the previous data base changes that have been made.

"query 3 c:" D"elete c:" C"hanges (to data base) ok:" C.R

"query 3 c:"

All of the previous changes have been deleted. The user can assure that this has been done by attempting to display the platform Cronkite.

LIST COMMAND - This command is used to obtain a listing current values for a particular field. It is of some value to the forgetful user and is particularly useful for determining legal field values when adding or changing platforms or characteristics. The command's alternatives are:

Cargo (type)	Flag	Synonym
Category	Label	Type
Class	Opcon	
Doctor	Plat	

"query 3 c:" ΔLI"st c:" C"urrent (values for field) c:" O"pcon ok:" C/R

TG77.3	CTG77.3	TG39.1	TU76.3.2	CTU76.3.2
TU76.3.1	CTU76.3.1	TG76.3	CTG76.3	TG37.7
CTG37.3	TG76.2	CTG76.2	CTF76	TU77.1.3
TU77.1.2	CTU77.1.2	CTG77.1	TU77.2.3	TU77.2.1
CTU77.2.1	CTG77.2	CTF77	TU33.7.2	CTU33.7.2
CTG33.7	TG33.2	CTG33.2	CTF33	TU36.3.1
TG36.2	CTG36.2	CTF36	TG37.2	CTG37.2
TG37.1	CTG37.1	CTF37		

IV. MODIFICATION OF HUMAN INTERFACE

Query 3 offers the capability of modifying human interface with the system through the "alter" command. This capability is a result of Query 3's NLS base and demonstrates the versatility of that system. The alter command is capable of eliminating the herald (Query 3) and the prompts (c:, ok/c:, t/⌐a¬:). Additionally, the command can increase user feedback by activating the search trace and expanding the search criteria.

Elimination of heralds and prompting is, of course, recommended for the experienced user only. However, the user feedback provided by the search trace and expanded search criteria may assist the inexperienced user in gaining an understanding of system operation. The following examples are presented to demonstrate the operation of the alter command.

"query 3 c:" ΔAL"ter c:" P"rompting (level to) c:"

N"one ok:" C/R

"query 3"

Notice that the prompt "c:" has been eliminated. Other prompts, such as "ok:" and t/¯a¯/:" have also been eliminated. Next, the herald query 3 will be altered.

"query c" ΔAL"ter" H"erald (to)" S"hort" C/R

"q"

All prompting has been eliminated and the herald has been reduced to the single letter "q". Now, the expanded search criteria and the search trace will be activated.

"q" ΔAL"ter" T"race (of searches to) O"n C/R

"q" ΔAL"ter" E"xpansion (of search criteria to)"

O"n" C/R

"q"

The modifications to human interface have been completed. The following examples demonstrate the results of this alteration.

"q" F"ind (all)" P"latforms (with)" T"ype" E"qual (to)"

CV C/R

A"nd" F"lag" E"qual (to)" US C/R C/R

"(display format?)" S"hort" C/R

"locating one or more platforms with type=cv and flag=us"

"independence us cv 20-06n 155-23w"

"Saratoga us cv 21-49s 122-02e"

"q" S"how" ΔAC"tive (surveillance threats to craft)"

SARATOGA C/R

"(past, present, or future?)" P"resent" C/R

"(display format?)" T"abular" C/R

"locating single craft with name=saratoga and class defined and ptp defined.
locating all classes with name=forrestal.
locating one or more craft with active surveillance threat to saratoga, i.e. (class=bear and within 250nm) or (class=bison and within 250nm) or (class=charlie and within 30nm) or (class=delta and within 35nm) or (class=echo ii and within 35 nm) or (class=foxtrot and within 40nm) or (class=kashin and within 125 nm) or (class=kazbek and within 75nm) or (class=kresta ii and within 175nm) or (class=kuril and within 175nm) or (class=kynda and within 175nm) or (class=may and within 240nm) or (class=moskva and within 175nm) or (class=wisla and within 150nm) or (class=okean and within 210nm) or (class=sverdlov and within 175nm) or (class=victor and within 40nm) or (class=yankee and within 40nm).
detailed search of 52 records is 0% complete; 0 records found so far.
detailed search of 52 records is 28% complete; 0 records found so far.
detailed search of 52 records is 65% complete; 1 records found so far.
"kronstadt"
"q"

The alter command has provided the user with two major aids to human interface in the above examples. First, the elimination of prompts and heralds has caused the queries to appear as close approximations of English sentences. Second, the activation of expanded search criteria and trace gives the user feedback concerning system logic and operation.

The alter command may also be used to return modified parameters to their original settings. This may be accomplished individually by parameter or in a global manner.

"q" ΔAL"ter" A"ll (parameters to original settings)" C/R

"query 3 c:"

The system has now been returned to its original condition.

BIBLIOGRAPHY

1. Augmentation Research Center, NLS-8 Glossary, 1975.

2. Augmentation Resources Center, NLS Text Book, pp. 1-1-5-18, 1979.

3. Defense Communications Agency Report NIC 45601, ARPANET Resource Handbook, October 1978.

4. Dock, V. T., and others, MIS A Managerial Perspective, Science Research Associates, Inc., p. 236, 1977.

5. George, W. A., Tactical Air Threat Query System Demonstration, M.S. Thesis, Naval Postgraduate School, Monterey, 1979.

6. Holg, C., Joy Of TENEX; 3... The Basics, USC/Information Sciences Institute, 1976.

7. Holg, C., Joy Of TENEX; 4...Some Of The Refinements, USC/Information Sciences Institute, 1977.

8. Kroenke, D., Data Base Processing, Science Research Associates, Inc., pp. 1-17, 1977.

9. Lucas, H. C., Information Systems Concepts For Management, McGraw-Hill, pp. 167-190, 1978.

10. Stanford Research Institute, Extensions To Query 2 Part I, 1976.

11. Information obtained by conversations with Dr. John Schill at the Naval Ocean Systems Center, San Diego, California.

12. Information obtained from Query 3 Software Documentation located online at Advanced Research Projects Agency Network Computer Host 116.

INITIAL DISTRIBUTION LIST

	No. Copies
1. Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2. Library, Code 0142 Naval Postgraduate School Monterey, California 93940	2
3. Professor John M. Wozencraft, Code 74 Chairman, C^3 Academic Group Naval Postgraduate School Monterey, California 93940	3
4. Professor G.K. Poock, Code 55Pk Department Of Operations Research Naval Postgraduate School Monterey, California 93940	35
5. Lt. Col. R.J. Roland, USAF, Code 52Rl Department Of Computer Science Naval Postgraduate School Monterey, California 93940	4
6. Lt. Col. F.R. Murphy, USAF, Code 39 C^3 Curricular Officer Naval Postgraduate School Monterey, California 93940	1
7. Department Chairman, Code 55 Department Of Operations Research Naval Postgraduate School Monterey, California 93940	1
8. Professor D.R. Barr, Code 55Bn Department of Operations Research Naval Postgraduate School Monterey, California 93940	1
9. Mr. R. Brandenburg Naval Ocean Systems Center Code 8321 San Diego, California 92152	1
10. Dr. M. Denicoff, Code 437 Office Of Naval Research 800 N. Quincy Street Arlington, Virginia 22217	1

11. LCDR A.J. Dietzler — 1
Advanced Research Projects Agency
Information Processing Techniques Office
1400 Wilson Boulevard
Arlington, Va. 22209

12. Professor D.P. Gaver, Code 55Gv — 1
Department of Operations Research
Naval Postgraudate School
Monterey, California 93940

13. Mr. H. Miller — 1
Naval Electronics Systems Command
PME-108-3
Washington, D.C. 20360

14. Mr. D. Norman, Code 0141 — 1
Manager of Operations
W.R. Church Computer Center
Naval Postgraudate School
Monterey, California 93940

15. Asso. Professor S.H. Parry, Code 55Py — 1
Department of Operations Research
Naval Postgraduate School
Monterey, California 93940

16. CDR. Don Lesemann — 1
Naval Ocean Systems Center
Code 832
San Diego, California 92152

17. Professor G.A. Rahe, Code 52Ra — 1
Department of Computer Science
Naval Postgraduate School
Monterey, California 93940

18. Assoc. Professor F.R. Richards, Code 55Rh — 1
Department of Operations Research
Naval Postgraduate School
Monterey, California 93940

19. Dr. Bob Kahn — 1
Advanced Research Projects Agency
Information Processing Techniques Office
1400 Wilson Boulevard
Arlington, Va. 22209

20. Dr. J. Schill — 1
Naval Ocean System Center
Code 8123
San Diego, California 92152

21. Professor N.F. Schneidewind, Code 54Ss — 1
Department of Administrative Sciences
Naval Postgraduate School
Monterey, California 93940

22. Mr. C.C. Stout — 1
Naval Electronics Systems Command
Code 330
2511 Jefferson Davis Highway
Arlington, Va. 20360

23. Dr. M. Tolcott, Code 455 — 1
Office of Naval Research
800 N. Quincy Street
Arlington, Va. 20360

24. CAPT R.D. Yingling, USAF, Code 62Ya — 1
Department of Electrical Engineering
Naval Postgraduate School
Monterey, California 93940

25. Mr. Gary Martins — 1
Rand Corporation
1700 Main St.
Santa Monica, California 90406

26. Mr. Richard Bisbey — 1
Information Sciences Institute
University of Southern California
4676 Admiralty Way
Marina Del Rey, California 90291

27. Dr. R. Kolb — 1
Naval Ocean Systems Center
Code 824
San Diego, California 92152

28. CDR. Richard Meinhold — 1
CINCPACFLT
P.O. Box 6
Pearl Harbor, Hawaii 96860

29. Dr. Earl Sacerdoti — 1
Stanford Research Institute
Artificial Intelligence Center
333 Ravenswood Avenue
Menlo Park, California 94025

30. Dr. Daniel Sagalowicz — 1
Stanford Research Institute
Artificial Intelligence Center
333 Ravenswood Avenue
Menlo Park, California 94025

31. Dr. Howard Morgan 1
Wharton School
University of Pennsylvania
Philadelphia, Pennsylvania 19104

32. Greg Gibbons 1
System Exploration Inc.
419 Webster St.
Monterey, California 93940

33. Dr. Norton R. Greenfeld 1
Bolt Beranek & Newman Inc.
50 Moulton Street
Cambridge, Mass. 02138

34. Dr. Scott Randall 1
Decisions And Design, Inc.
8400 Westpark Drive, Suite 600
McLean, Virginia 22101

35. LCDR L.J. Faneuf, III, USN 1
Office Of The Chief Of Naval Operations
OP-942
Navy Department
Washington, D.C. 20350

Thesis
F217
c.1

190130

Faneuf

Query 3; a data base inquiry system description and user's tutorials.